FORM PTO-1390
(REV. 5-93)

U.S. DEPARTMENT OF COMMERCE
PATENT AND TRADEMARK OFFICE

TRANSMITTAL LETTER TO THE UNITED STATES DESIGNATED/ELECTED OFFICE (DO/EO/US)

ATTORNEY'S DOCKET NUMBER: ASA-1016

U.S. APPLICATION NO (if known, see 37 CFR 1.5): 09/890703

International Application No	International Filing Date	Priority Date Claimed
PCT/JP99/00493	February 5, 1999	

Title of Invention: DATA PROCESSING SYSTEM

Applicant(s) for DO/EO/US: M. NAKAGAWA et al (see attached)

Applicant herewith submits to the United States Designated/Elected Office (DO/EO/US) the following items and other information:

1. [X] This is a FIRST submission of items concerning a filing under 35 U.S.C. 371.
2. [ ] This is a SECOND or SUBSEQUENT submission of items concerning a filing under 35 U.S.C. 371.
3. [X] This express request to begin national examination procedures (35 U.S.C. 371(f)) at any time rather than delay examination until the expiration of the applicable time limit set in 35 U.S.C. 371(b) and PCT Articles 22 and 39(1).
4. [X] A proper Demand for International Preliminary Examination was made by the 19th month from the earliest claimed priority date.
5. [X] A copy of the International Application as filed (35 U.S.C. 371(c)(2))
   a. [ ] is transmitted herewith (required only if not transmitted by the International Bureau).
   b. [X] has been transmitted by the International Bureau.
   c. [ ] is not required, as the application was filed in the United States Receiving Office (RO/US).
6. [X] A translation of the International Application into English (35 U.S.C. 371(c)(2)).
7. [ ] Amendments to the claims of the International Application under PCT Article 19 (35 U.S.C. 371(c)(3))
   a. [ ] are transmitted herewith (required only if not transmitted by the International Bureau).
   b. [ ] have been transmitted by the International Bureau.
   c. [ ] have not been made; however, the time limit for making such amendments has NOT expired.
   d. [ ] have not been made and will not be made.
8. [ ] A translation of the amendments to the claims under PCT Article 19 (35 U.S.C. 371(c)(3)).
9. [ ] An oath or declaration of the inventor(s) (35 U.S.C. 371(c)(4)).
10. [ ] A translation of the annexes to the International Preliminary Examination Report under PCT Article 36 (35 U.S.C. 371(c)(5)).

Items 11. to 16. below concern other document(s) or information included:

11. [X] An Information Disclosure Statement under 37 CFR 1.97 and 1.98.
12. [ ] An assignment document for recording. A separate cover sheet in compliance with 37 CFR 3.28 and 3.31 is included.
13. [X] A FIRST preliminary amendment.
    [ ] A SECOND or SUBSEQUENT preliminary amendment.
14. [ ] A substitute specification.
15. [ ] A change of power of attorney and/or address letter.
16. [X] Other items or information:

[X] THIS APPLICATION IS BEING FILED WITHOUT AN EXECUTED DECLARATION AND POWER OF ATTORNEY, WHICH WILL BE FILED LATER.

[x] LIST OF INVENTORS' NAMES AND ADDRESSES.

JC17 Rec'd PCT/PTO 02 AUG 2001

U.S. Application No (if known, see 37 CFR 1.5)	International Application No	Attorney's Docket Number
09/890703	PCT/JP99/00493	ASA-1016

17. [X] The following fees are submitted:

	CALCULATIONS	PTO USE ONLY
Basic National Fee (37 CFR 1.492 (a)(1)-(5)):		
Search Report has been prepared by the EPO or JPO $860.00		
International preliminary examination fee paid to USPTO (37 CFR 1.482) $690.00		
No international preliminary examination fee (37 CFR 1.482) but international search fee paid to USPTO (37 CFR 1.445 (A)(2)) $710.00		
Neither international examination fee (37 CFR 1.482) nor international search fee (37 CFR 1.445(A)(2)) paid to USPTO $1000.00		
International preliminary examination fee paid to USPTO (37 CFR 1.482) and all claims satisfied provisions of PCT Article 33(2) to (4) $100.00		
ENTER APPROPRIATE BASIC FEE AMOUNT =	$ 860.00	
Surcharge of $130.00 for furnishing the oath or declaration later than [ ] 20 [X] 30 months from the earliest claimed priority date (37 CFR 1.492(e)). +	$ 130.00	

Claims	Number Filed	Number Extra	Rate		
Total	28 -20 =	8	x $18.00	$ 144.00	
Independent	6 - 3 =	3	x $80.00	$ 240.00	
Multiple dependent claim(s) (if applicable)			+ $270.00	$ 0.00	
TOTAL OF ABOVE CALCULATIONS =				$ 1,374.00	
Reduction by 1/2 for filing by small entity, if applicable. Verified Small Entity statement must also be filed. (Note 37 CFR 1.9, 1.27, 1.28).				$ 0.00	
SUBTOTAL =				$ 1,374.00	
Processing fee of $130.00 for furnishing the English translation later than [ ] 20 [ ] 30 months from the earliest claimed priority date (37 CFR 1.492(f)). +				$ 0.00	
TOTAL NATIONAL FEE =				$ 1,374.00	
Fee for recording the enclosed assignment (37 CFR 1.21(h)). The assignment must be accompanied by an appropriate cover sheet (37 CFR 3.28, 3.31). $40.00 per property. +				$ 0.00	
TOTAL FEES ENCLOSED =				$ 1,374.00	
				Amount to be: Refunded	$
				Charged	$

a. [X] **A check in the amount of $** 1,374.00 **to cover the above fees is enclosed.**

b. [ ] **Please charge my Deposit Account No. 50-1417 in the amount of $ ______ to cover the above fees. A duplicate copy of this sheet is enclosed.**

c. [X] **The Commissioner is hereby authorized to charge any additional fees which may be required, or credit any overpayment to Deposit Account No. 50-1417. A duplicate copy of this sheet is enclosed.**

Note: Where an appropriate time limit under 37 CFR 1.494 or 1.495 has not been met, a petition to revive (37 CFR 1.137(a) or (b)) must be filed and granted to restore the application to pending status.

SEND ALL CORRESPONDENCE TO:

MATTINGLY, STANGER & MALUR, P.C.
1800 Diagonal Rd., Suite 370
Alexandria, Virginia 22314
(703) 684 -1120

24956
PATENT TRADEMARK OFFICE

Signature

John R. Mattingly
Name

30,293
Registration Number

ASA-1016

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of

M. NAKAGAWA et al

Serial No.

Filed: August 2, 2001

For: DATA PROCESSING SYSTEM

PRELIMINARY AMENDMENT

Commissioner for Patents
Washington, D.C. 20231

Sir:

Prior to examination, please amend the above-identified application as follows.

IN THE CLAIMS

Rewrite claims 3, 4, 15 and 28 follows:

3. (Amended) A data processing system as set forth in claim 1, wherein said entire distribution based on each of said one-dimensional Gaussian distributions is represented by 2^N numeric values, and the quantized value of said feature component correspond to upper N bits of said values.

4. (Amended) A data processing system as set forth in claim 1, wherein said data processor repetitively refers to said numeric value table for each feature component to compute the values of the multi-dimensional Gaussian distributions, and repetitively computes the values of the multi-dimensional Gaussian distributions by a predetermined number of times to compute the output probability represented by the mixture multi-dimensional Gaussian distribution.

15. (Amended) A data processing system as set forth in claim 7, wherein said data processor linearly quantizes all feature components of a feature vector, computes a feature offset from a first location of the extracted intermediate table on the basis of a product of said quantized value and an address amount of a single array element of said X-direction array, and thereafter refers to the intermediate table on the basis of said access pointer and feature offset for each multi-dimension mixture Gaussian distribution to refer to the numeric value table.

28. (Amended) A data processing system as set forth in claim 1, having a battery for supplying an operational power, and wherein said data processor operates on said battery as its operating power source and has a power consumption of 1W or less.

REMARKS

Examination is requested.

Respectfully submitted,

John R. Mattingly
Registration No. 30,293
Attorney for Applicant(s)

MATTINGLY, STANGER & MALUR
1800 Diagonal Road, Suite 370
Alexandria, Virginia 22314
(703) 684-1120
Date: August 2, 2001

MARKED UP VERSION OF REWRITTEN CLAIMS

3. (Amended) A data processing system as set forth in claim 1 [or 2], wherein said entire distribution based on each of said one-dimensional Gaussian distributions is represented by 2^N numeric values, and the quantized value of said feature component correspond to upper N bits of said values.

4. (Amended) A data processing system as set forth in claim 1 [or 2], wherein said data processor repetitively refers to said numeric value table for each feature component to compute the values of the multi-dimensional Gaussian distributions, and repetitively computes the values of the multi-dimensional Gaussian distributions by a predetermined number of times to compute the output probability represented by the mixture multi-dimensional Gaussian distribution.

15. (Amended) A data processing system as set forth in claim 7 [or 8], wherein said data processor linearly quantizes all feature components of a feature vector, computes a feature offset from a first location of the extracted intermediate table on the basis of a product of said quantized value and an address amount of a single array element of said X-direction array, and thereafter refers to the intermediate table on the basis of said access pointer and feature offset for each

multi-dimension mixture Gaussian distribution to refer to the numeric value table.

28. (Amended) A data processing system as set forth in claim 1 [or 7], having a battery for supplying an operational power, and wherein said data processor operates on said battery as its operating power source and has a power consumption of 1W or less.

28/PRTS

JC17 Rec'd PCT/PTO 02 AUG 2001

E6055

09/890703 116/28

DESCRIPTION

DATA PROCESSING SYSTEM

TECHNICAL FIELD

The present invention relates to a voice recognition technique based on continuous mixture hidden Markov model (HMM) using mixture Gaussian distribution Ns and more particularly, to a technique for calculating an output probability therefor. More particularly, the present invention concerns a technique for effective use, e.g., in a portable information terminal unit which has a data processor driven on a battery to perform arithmetic operations for voice recognition.

BACKGROUND ART

A hidden Markov model is a state transition model represented by a Markov process (stochastic process given only by a state at a time point (t+1) o at a time point 'n'). The hidden Markov model can be applied to a speech recognition technique. An outline of the speech recognition technique will be explained in plain English. A voice to be recognized is divided into partial sections (frames) such as 10ms, and a feature vector such as a frequency spectrum is extracted for each frame. At this time, a chain of the partial sections of the speech to be recognized is regarded as a chain of states of the frames. If the

respective states are determined so that a voice source approximated as a feature vector is assigned to each state, then the speech recognition will be realized. To this end, an output probability representative of a likelihood with which each state is comparable to a feature vector for each voice source as well as a state transition probability with which the current state is changed to a state adjacent thereto can be employed so that a chain of states when the sum of products of the output probabilities and state transition probabilities for the states is maximum, is regarded as a speech recognition result. The amount of computations necessary for the products of the state transition probabilities and output probabilities for frames of each pattern estimated from a column of the feature vectors, become enormous. In particular, the output probability is given by a mixture multi-dimensional Gaussian distribution. The mixture multi-dimensional Gaussian distribution has stochastic distributions of elements including age and sex for a phoneme, e.g., of "あ[a]", each of the stochastic distributions has multi-dimensional Gaussian distributions corresponding to the orders of the feature vectors, and each of the multi-dimensional Gaussian distributions is a probability distribution corresponding to a composite of one-dimensional Gaussian distributions. Accordingly the larger the number of mixture multi-dimensional Gaussian distributions and the number of orders of the feature

vectors is the more the time for computations of the output probabilities is required. According to trial computation of the inventors of the present application, the computation load of output probabilities is estimated to be as enormous as 50-80% of the overall speech recognition processing amount.

For the purpose of reducing the computation amount of output probabilities, it is effective to reduce a range to be calculated in the mixture multi-dimensional Gaussian distribution. For example, there can be employed a method for associating feature vectors with several standard patterns (vector quantization) and defining an output probability for each pattern. In this case, a feature space is divided into partial regions and the partial regions are associated with distributions to be calculated. In this connection, vector quantization can be used for the correlation between the feature vector and partial region. The vector quantization is a method wherein a finite number of representative vectors on the feature space are considered and a given point on the feature space is expressed by approximation in terms of a representative vector closest to the point. Several effective methods of such vector quantization have been suggested. However, these methods are fundamentally based on the fact that a representative vector corresponding to a point having the smallest distance therefrom is selected. Thus the computation amount is

very small when compared with the computation amount for the mixture distribution, but the computation load is not small enough.

It is also possible to convert part of the computations to a table to realized high speed computation. Even in this case, the table can be constituted in the form of the vector quantization. However, when the table is vector quantized and associated with the output probability, a quantization error becomes large and thus a recognition performance is deteriorated.

In order to avoid this, it is considered to resolve the computation into feature dimensional computations, divide each of the feature dimensions into standard Gaussian distribution patterns, and then convert computation results to a table. Such a method is employed for scalar quantization. As the scalar quantization, for example, there is a technique for converting a single Gaussian distribution to a table. In this case, unlike the vector quantization, the quantization error becomes very small.

Employable as the scalar quantization is a non-linear scalar quantization. That is, this quantization method is intended to reduce the number of types in data tables, since highest one of the feature orders of feature vectors is several tens of dimension and thus it is inefficient to convert all single Gaussian distributions to tables. In the scalar

quantization of the mixture Gaussian distribution, a function for each dimension is a single one-dimensional normal distribution (single Gaussian distribution). Thus when the single Gaussian distribution is employed, the computation of the output probabilities can be simplified. The correlation of the one-dimensional normal distributions possibly different in feature orders and mixtures can be defined when an average and dispersion of each distribution are known. In order to determine the correlation, a parameter is calculated for each feature dimension and the calculated parameter and the feature component of the feature vector are used to access a numeric value table of the one-dimensional normal distribution typically provided. Such a technique for reducing the amount of mixture HMM computations by using the non-linear scalar quantization and accessing the numeric value table is disclosed, for example, "ON THE USE OF SCALAR QUANTIZATION FOR FAST HMM COMPUTATION", ICASSP 95, pp. 213-216.

In this technique, however, it is always required to perform parameter computation for each feature component for the table access. In addition, even in the table look-up, access using such computed parameter is not always a continuous array access to the table. For this reason, address computation for the table look-up also requires multiplication and addition for each look-up.

The realization of the numeric value table look-up while eliminating the need for such troublesome parameter computation is , for example, to employ linear scalar quantization using general linear quantization. In other words, features are quantized as spaced uniformly. For example, when a data table for a single Gaussian distribution is divided into the n-th power of 2 (2^N) parts for easy quantization, quantization can be easily realized by extracting upper N bits of the feature component. In the linear scalar quantization, since a representative point is fixed, linear scalar quantization for the mixture multi-dimensional Gaussian distribution is required only once for each frame. In other words, it is only required to perform the quantization once for each feature dimension. Since the representative value can be used as an index as it is, a difference (which will be also referred to as the offset, hereinafter) between lead and desired addresses in the numeric value table corresponds to (index times data length) which is also common in all distributions. Thus such computation is required to be executed merely once for one frame. And since access to a necessary numeric value table is computed in the form of a sum of the address of each numeric value table and the offset common to all the feature components, the access is eventually carried out with one addition computation and two loads (of lead address and numeric value data).

In computation of the output probability of the mixture Gaussian HMM, it is important to reduce the amount of computations for the single Gaussian distribution (including a logarithmic type). The computation of the single Gaussian distribution for each feature component corresponds to the heaviest part of the computation load and the number of computations is represented by (the number of all models x the number of mixtures x the number of feature dimensions). For this reason, a slight increase in computation cost results directly in an increase in the entire computation amount. Since perfectly no computation of the linear scalar quantization is generated except for the table access, the linear scalar quantization is highly excellent from the viewpoint of computation efficiency.

The linear scalar quantization is very high in speed from the viewpoint of computation efficiency, but a numeric value table is required for each distribution with respect to a fixed representative point. Accordingly the linear scalar quantization has a big problem that the number of numeric value tables or the amount of data become enormous. Further, when the parameters (average and dispersion) of the mixture Gaussian distribution are modified for speaker adaptive processing or noise adaptive processing, this involves correspondingly increased massive amount of computations. Similarly even modification of the

numeric value table requires a massive amount of processing.

When the non-linear scalar quantization is employed, as has been explained above, this requires an enormous amount of computations for the numeric value table look-up; whereas, when the linear scalar quantization is employed, the numeric value table look-up can be made efficient but this still requires a very large number of numeric value tables. Even in either case, these methods are too enormous in processing amount to practical in a portable information terminal device or in a data processing system having a strict demand of low cost such as a battery-driven data processing system.

It is therefore an object of the present invention to provide a data processing system which can compute the output probability of an HMM at a high speed and can flexibly cope with model modification such as speaker adaptive processing or environmental adaptive processing, and also to provide a method for computing the output probability of a mixture Gaussian HMM.

Another object of the present invention is to provide a data processing system which can realize high-speed computation of the output probability and high-speed processing of a modification in a multi-dimensional Gaussian distribution caused by an adaptation even when the system is a portable

information terminal device, a data processing system having a relatively low computation processing ability such as a battery-driven type, or a data processing system requiring a strict low cost demand.

The above and other objects and novel features of the present invention will become clear from the following description with reference to the attached drawings.

DISCLOSURE OF INVENTION

<<Variable Mapping Based On Intermediate Table>>

In a mixture Gaussian HMM, an output probability is expressed by a function (Equation (2)) with respect to mixture multi-dimensional Gaussian distribution. For example, a mixture multi-dimensional Gaussian distribution is a sum of multi-dimensional Gaussian distributions, and each multi-dimensional Gaussian distribution is a product of one-dimensional Gaussian distributions. A feature component is a component of a feature vector as an observation system of a speech to be recognized. A dispersion and average of the one-dimensional Gaussian distributions for each feature component are inherent in each feature component. When numeric values of various one-dimensional Gaussian distributions are tabulated, the numeric value tables of the one-dimensional Gaussian distributions will not prepared respectively one for each feature component. Intermediate tables 301, 401

are provided. That is, a numeric value table 1052 has numeric values of a plurality of one-dimensional Gaussian distributions having representative dispersions and averages stored therein. Linear scalar quantization is employed for the feature component and its quantized value is used as an index to look up or refer to information on the intermediate table. When the intermediate table is provided for each feature component, the intermediate table contains address information indicative of the locations of numeric values on the numeric value table relating to the one-dimensional Gaussian distribution corresponding to a required dispersion and average. When the dispersion and average of the one-dimensional Gaussian distribution is modified by adaptation, contents of the intermediate table are rewritten according to the locations of the one-dimensional Gaussian distribution corresponding to the modified dispersion and average.

A global table 400 can be formed commonly to the respective feature components so that the intermediate table is extracted from the global table for use. The global table, as exemplified in Fig. 17, has storage zone arrays arranged in a matrix in X and Y directions, ones of which arrays in the X direction each contain address information indicative of the locations of numeric values of corresponding one-dimensional Gaussian distributions corresponding on a numeric value table, dispersions of the one-dimensional

Gaussian distributions relating to the X-direction arrays are made different from each other, and averages thereof are unified, e.g., at the centers of the distributions. The value of the dispersion of the one-dimensional Gaussian distributions is taken into consideration to select an Y-direction arrays in the global table; whereas the value of the average of the one-dimensional Gaussian distributions is taken into consideration to select the first location in the X direction. The larger the value of the average is the more toward the X direction the first location is shifted. An intermediate table starting with the first position of the X direction can be extracted on the basis of the Y direction location of the global table and the first location of the X direction thereof. For an access to the extracted intermediate table, as in the aforementioned case, the quantized value of the feature component is used as an offset from the first location. When it is desired to modify only the dispersion of the one-dimensional Gaussian distributions due to adaptation, it is only required to change the Y direction location at the time of extracting the intermediate table. When it is desired to modify only the average of the one-dimensional Gaussian distributions due to adaptation, it is only required to change the first location of the X direction at the time of extracting the intermediate table. The first address of the intermediate table to

be extracted for each feature component may be specified by an access pointer P0 to Pn. The value of the access pointer may be previously computed according to the dispersion σ and average μ. Upon adaptation, the value of the access pointer can be previously modified according to the modification of the dispersion and average. The access pointers for the respective feature components may be previously set collectively in an access pointer table 420.

As mentioned above, for the purpose of coping with the modification of the average and dispersion while avoiding complex parameter computation to refer to the numeric value table for each feature component, the linear scalar quantization was employed. Further, for the purpose of controlling a pattern of access to the numeric value table corresponding to the feature component subjected to the linear quantization, the intermediate table was employed. By inserting the intermediate table intended for index transformation to enable a mapping relation between the linearly-quantized feature component and numeric value table, the system can easily cope with the modification of the dispersion and average caused by the adaptation. That is, the aforementioned arrangement using the global table can cope with such modification of the dispersion and average caused by the adaptation merely by modifying the access pointer. Put another way, reduction in the amount of data can be realized as in

the non-linear scalar quantization while ensuring the high-speed look-up of the numeric value table based on the linear scalar quantization, by combining the intermediate tables intended for the linear scalar quantization and index transformation.

<<Efficiency Increased By Typification And Commonality Of Index Transformation>>

When the aforementioned arrangement is implemented in a simple manner, rewriting of the numeric value table will not take place but instead, rewriting of the intermediate table, etc. will take place. In order to cope with this problem, first, (a) such an arrangement is employed that an intermediate transformation pattern based on the typification of the index transformation is previously computed. That is, the speaker adaptive processing or environmental adaptive processing is carried out by modification or change of the average and dispersion of the Gaussian distribution. When the average and dispersion pattern is typified and previously held, table modification cost can be minimized. Second, (b) the arrangement is simplified by commonly using the intermediate table. That is, in the aforementioned method, it is assumed to have an intermediate table for each mixture distribution with respect to each HMM, which means that, so long as there is a single table which covers all transformation tables, the function of the

intermediate table can be realized by listing an access location (of each mixture distribution for each HMM) in the table. In this case, the speaker and environmental adaptive processing can be sufficiently realized only by modifying the above access location.

<<Selection Of Computation Distribution By Intermediate Table>>

In the computation of the mixture Gaussian distribution, reduction of the distribution to be computed is a valid method for increasing the computation speed. In the present invention, the computation can be simplified by providing the intermediate table with a distribution selecting function. In general, a multi-dimensional Gaussian distribution is represented by a product of one-dimensional Gaussian distributions in each feature dimension. By inserting an evaluation for each one-dimensional Gaussian distribution in the intermediate table, the frequency of useless look-up operation to the numeric value table can be reduced, thus realizing the distribution reducing function.

<<Data Processing System>>

In a data processing system in accordance with an embodiment of the present invention, the data processor 103 can refer to intermediate tables 301 and 302 and the numeric value table 1052 to compute an

output probability represented by a mixture multi-dimensional Gaussian distribution for HMM speech recognition to the feature vector. The numeric value table 1052 has a region 1052E containing numeric values of distributions based on a plurality of types of one-dimensional Gaussian distributions, and the intermediate tables 301 and 302 have regions 301E and 302E containing address information indicative of locations of values of the numeric value table corresponding to quantized values in a region selected based on the linearly-quantized values of values of feature components of the feature vector respectively. And the data processor linearly quantizes the value of the feature component, selects an intermediate table based on the access pointers(P0 to Pn in a table 310) of the feature component, acquires address information based on the intermediate table selected on the basis of the linearly-quantized value, refers to the numeric value table using the acquired address information, and computes the output probability based on the value referred to the numeric value table.

The above data processing system may have a region for formation of the access pointer table 310 wherein the access pointers for the feature components are listed in each multi-dimensional Gaussian distribution of the mixture multi-dimensional Gaussian distribution, and may be arranged so that the data processor selects an intermediate table using the

access pointer of the access pointer table.

With regard to the quantization, assuming that the entire distributions based on the one-dimensional Gaussian distributions are expressed by 2^N numeric values, then the quantized value of the feature component corresponds to upper N bits of the value. This means the quantization can be realized only through the shift operation of the feature component.

The data processor can repeat the referring operation to the numeric value table for every feature component to compute the value of the multi-dimensional Gaussian distribution, and can repeat the computing operation of the value of the multi-dimensional Gaussian distribution by a constant number of times to compute an output probability represented by the mixture multi-dimensional Gaussian distribution.

Distance information for distribution reduction can be previously contained in the intermediate table. The intermediate table has a region E1 which contains the address information for a range of the dispersion multiplied by a plurality of times with the average location as a start point of the one-dimensional Gaussian distribution as a reference of the numeric value table. The intermediate table has a region E2 which contains distance information from the average outside of the region E1. The data processor can repeat the referring operation to the numeric value table for every feature component to compute the value

of the multi-dimensional Gaussian distribution, can accumulate it when information referred to by the intermediate table is the distance information, and can stop the operation for the multi-dimensional Gaussian distribution when the accumulated value exceeds a predetermined value.

A region E3 containing a fixed value (such as a value "0") outside the distance information is provided as other distribution reduction information in the intermediate table, so that, when referring to the fixed value from the intermediate table, the data processor can stop the operation for the multi-dimensional Gaussian distribution being currently processed.

The data processing system may comprise, for example, a portable information terminal device 120 which uses a battery 121 as an operating power source. The device driven by the battery is strictly required to have a low power consumption and can reduce the aforementioned computation load of the output probability, so that, even when the data processor has a power consumption of 1W or less, the device can perform speech recognizing operation at a practically high speed.

<<Data Processing System Using Global Table>>

In a data processing system specialized in using a global table, the data processor 103 can refer

to the global table 400 and numeric value table 1052 to compute an output probability represented by the mixture multi-dimensional Gaussian distribution for HMM speech recognition of the feature vector. The numeric value table 1052 has the region 1052E which contains the numeric value of each distribution based on a plurality of types of one-dimensional Gaussian distributions having different dispersions, the global table 400 has a region 400E which contains a plurality of sets of X-direction arrays of each distribution in the numeric value table, and the X-direction arrays contain address information indicating that the presence of the value of the numeric value table corresponding to its linearly quantized value at the location selected based on the quantized value of the value of the feature component of the feature vector. The data processor can linearly quantize the value of the feature component, can extract the intermediate tables 401 and 402 from the global table according to the value of the access pointer (P0 to Pn in Fig. 38) of each feature component when dispersion is taken into consideration in Y direction selection of the plurality of sets of X-direction arrays and when average is taken into consideration in determination of the first location of the X-direction arrays, can acquire the address information on the basis of the linearly-quantized value with the first location of the extracted intermediate table as a start point, can

refer to the numeric value table using the acquired address information, and can compute the output probability on the basis of the value referred to from the numeric value table.

The data processor can extract the intermediate table with use of the access pointer (P0 to Pn) of the access pointer table 420. The access pointer table is a table in which the access pointers for the feature components are arranged for the respective multi-dimensional Gaussian distributions of the mixture multi-dimensional Gaussian distribution.

When both or either one of the average and dispersion of the mixture multi-dimensional Gaussian distribution is changed by adaptive processing, the data processor is only required to correspondingly change the values of the access pointers of the access pointer table. It is unnecessary to change the contents of the global table per se.

When a plurality of sets of such access pointer tables are previously set, the data processor can identify the speaker and can use one of the access pointer tables according to the identified result.

The speaker identification may be realized on the basis of a state of a switch 1302SW for speaker identification. For example, in a data processing system such as a transceiver based on one-way speech, speaker identification can be realized as operatively associated with the change-over between send voice and

receive voices.

A management table 500 can be employed to link the speaker to the access pointer table. At this time, the data processor identifies the speaker on the basis of a comparison result between previously-registered identification feature information indicative of the feature of the speaker and an actually-analyzed speech feature result, and when the identified speaker is registered in the management table, the data processor refers to the access pointer table of the registered speaker.

The data processor can limit the number of speakers registerable in the management table to a fixed value, add information on use frequency for each of the registered speakers in the management table, and when the speech feature analyzed result indicates the registered speaker, can increment the use frequency of the registered speaker coinciding with the analyzed result and decrement the user frequencies of the speakers not coinciding with the analyzed result, and when the speech feature analyzed result indicates a speaker other than the registered speakers, can delete one of the registered speakers having the lowest use frequency from the management table and instead add the non-registered speaker to the management table.

Or the system may have a plurality of speech input channels each having such an access pointer table as mentioned, and the data processor may perform

parallel speech recognizing operations over the plurality of speech input channels independently using the access pointer tables.

The data processor can linearly quantize all the feature components of the feature vector, compute a feature offset from the first location of the extracted intermediate table on the basis of a product of the quantized value and an address amount of single array elements of the X-direction arrays, and thereafter can refer to the intermediate table for each multi-dimensional mixture Gaussian distribution on the basis of the access pointer and feature offset to refer to the numeric value table. As a result, the need for retrying to compute the feature offset for each mixture multi-dimensional Gaussian distribution can be eliminated.

A control program for computation of the output probability for speech recognition to be executed in the above data processing system can be provided to the data processing system via a computer-readable recording medium.

BRIEF DESCRIPTION OF DRAWINGS

Fig. 1 is a block diagram of an embodiment of a speech recognition system using a microcomputer;

Fig. 2 is a block diagram of an example of the microcomputer;

Fig. 3 is a flowchart for explaining the

entire schematic operations executed by the speech recognition system shown in Fig. 1;

Fig. 4 is a flowchart showing the summary of recognizing operations;

Fig. 5 is a diagram for explaining an example of HMM;

Fig. 6 is a diagram for explaining an example of left-to-right type HMM model;

Fig. 7 is a diagram for explaining three mixture two-dimensional Gaussian distributions as examples of mixture multi-dimensional Gaussian distribution;

Fig. 8 is a diagram of a two-dimensional feature space when taken along a section 1 in Fig. 7 and viewed from its side;

Fig. 9 is a diagram for explaining a relationship between a numeric value table and a one-dimensional normal distribution when linear scalar quantization is carried out;

Fig. 10 is an explanatory diagram for exemplifying the principle of linear scalar quantization;

Fig. 11 is a diagram for explaining an example of average and dispersion of a one-dimensional Gaussian distribution;

Fig. 12 is a diagram for explaining a one-dimensional Gaussian distribution having a different average and dispersion from those in Fig. 11;

Fig. 13 schematically shows a data structure of an intermediate table for distribution reduction;

Fig. 14 is a diagram for explaining an example of distance information for distribution reduction in the intermediate table;

Fig. 15 is a diagram for explaining an example of an array of distribution reduction information of the intermediate table for a single Gaussian distribution;

Fig. 16 is a flowchart exemplifying operation branching according to the value of the intermediate table;

Fig. 17 is a diagram for explaining an example of a global intermediate table;

Fig. 18 is a flowchart showing a detailed example of computation of an output probability;

Fig. 19 is a flowchart showing an example of modification of average and dispersion of a mixture Gaussian distribution in adaptive processing;

Fig. 20 is a flowchart generally showing an example of a processing procedure to determine the value of an intermediate table pointer corresponding to the dispersion and average of a Gaussian distribution modified by the adaptive processing in Fig. 19;

Fig. 21 shows an example of an outside view of a portable information terminal device to which a speech recognition system is applied;

Fig. 22 is an exemplary block diagram of the

portable information terminal device shown in Fig. 21;

Fig. 23 is a flowchart showing a detailed example of a processing procedure when noise adaptive processing is carried out with use of two microphones in the portable information terminal device;

Fig. 24 is a flowchart showing an example of a speech recognizing procedure in transceiver type speech of the portable information terminal device;

Fig. 25 is a flowchart showing an example of a speech recognizing procedure in separate type speech of the portable information terminal device;

Fig. 26 is a flowchart showing an example of a speech recognizing operation in a speech recognition system capable of performing speaker adaptive processing and noise adaptive processing;

Fig. 27 is a flowchart showing an example of a speech recognizing procedure to determine a registered speaker by his use frequency when speaker adaptive processing without teacher is executed;

Fig. 28 is a flowchart showing an example of a speech recognizing procedure to keep a fixed number of registered speakers by their use frequencies when the speaker adaptive processing of no teacher is executed;

Fig. 29 shows an example of a structure of a speaker management table relating to speaker management of identification information for the speaker adaptive processing;

Fig. 30 is a flowchart showing an example of operations of modifying and changing the structure of the speaker management table according to the frequency information;

Fig. 31 is a diagram showing an example of operation to a list newly exchanged in the speaker management table by initializing;

Fig. 32 is a diagram showing an example of operation to a list already present in the speaker management table;

Fig. 33 is a flowchart showing a processing procedure of Figs. 31 and 32;

Fig. 34 is a diagram for explaining the principle of two-microphone noise adaptive processing;

Fig. 35 is a diagram for explaining the principle of speech recognition in the transceiver type speech;

Fig. 36 is a diagram for explaining the principle of speech recognition in the separate type speech;

Fig. 37 is a diagram for explaining the principle of how to modify a table first address point according to the noise adaptive processing;

Fig. 38 is a diagram for explaining an example of the structure of an access pointer table for a global table included in an HMM parameter set;

Fig. 39 is a diagram for explaining an example of the structure of an access pointer table for

an intermediate table included in the HMM parameter set;

Fig. 40 is a diagram for explaining a table access technique for probability computation using a multi-dimensional Gaussian distribution;

Fig. 41 shows a relationship between access to the intermediate table and access to the numeric value table on a time series basis;

Fig. 42 shows an example of the numeric value table for a one-dimensional Gaussian distribution suitable when a microprocessor supporting floating point arithmetic operation is used; and

Fig. 43 shows an example of the numeric value table for the one-dimensional Gaussian distribution capable of coping with integer processing.

BEST MODE FOR CARRYING OUT THE INVENTION

<<Summary Of Speech Recognition Using Mixture Gaussian HMM>>

Explanation will first be made as to the basic contents of a speech recognition technique using a mixture Gaussian HMM.

Fig. 5 shows an example of an HMM. It will be appreciated from the drawing that the HMM is a state transition model represented by a Markov process (stochastic process given only by a state at a time point (t+1) or by a state at a time point "n").

In the speech recognition, these states are

regarded as two types of stochastic "voice source". In this connection, the expression " stochastic" means that, when the model is in such a state, it is not always that a predetermined voice is generated but there is a probability of generating various voice, which is generally called an output probability.

In the speech recognition, word and voice are represented by a model wherein a partial order relation is given to a state therebetween for their connection. More specifically, such a left-to-right type HMM as shown in Fig. 6 is used in many cases.

For instance, consider how to represent a word "あい[ai]" by a left-to-right type HMM. Assume that "あい[ai]" is "Word 1". And "あ[a]" is assumed to be represented by a state S1 and "い[i]" is to be represented by a state S2.

At this time, if "あ[a]" has one frame (e.g., 10ms) and "い[i]" has one frame (e.g., 10ms) at all times, then word "あい[ai]" will be expressed by a state transition of S1 to S2. In actuality, however, "あ[a]" having a varying length of time is followed by "い[i]" having a varying length of time.

In order to express such variations in time, a own state transition and a state transition adjacent thereto are expressed with a probability. Thus a voicing pattern of "あ[a]" having a continuation of **n** frames followed by "い[i]" a continuation of **m** frames can be expressed by probabilities (in the form of

occurrence probabilities of each pattern). This probability is a transition probability (state transition probability). In the case of the Word 1 in Fig. 6, a1(1,1) is a state transition probability with which the state S1 is changed to the same state S1, and a1(1,2) is a state transition probability with which the state S1 is changed to the next state S2 adjacent thereto.

With respect to the voice "あ[a]", its acoustic property varies depending on the type of speaker who is man, woman, old or young, with various statistical occurrence patterns. Thus when the output pattern of a feature vector in the state S1 representing the voice "あ[a]" is represented by a probability of one of such statistical occurrence patterns, voice patterns of various persons can be modeled. This stochastic representation is an output probability. In Fig. 6, the output probability of the Word 1 in the state S1 is represented by b11(y) and the output probability of the Word 1 in the state S2 is represented by b12(y).

As mentioned above, in order to "express variations of a word of various types of persons in time and acoustics" in HMM, person's voice process is modeled with probability and thus its estimation must be expressed with probability, as a matter of course. That is, given an observation sequence (analyzed result of an input voice), in models for expression of each

word, a probability (likelihood) that the observation sequence can be obtained is evaluated and one of the models having a highest likelihood (or a word meant by it) is output as a recognition candidate.

As mentioned above, in the HMM speech recognition, a model having the highest likelihood is output as a recognition candidate. For this reason, it is required to compute a likelihood for each model and thus to compute a product of a state transition probability and an output probability for each state. Thus an enormous amount of computation load as a whole is estimated. Such an enormous amount of computation is processed, for example, by a sort of dynamic programming called a Viterbi algorithm.

In the Viterbi algorithm, one (optimum path) of a plurality of state transition paths having a highest likelihood is selected and evaluation is carried out with the selected likelihood. The computation can be efficiently carried out according to an equation (1), which follows.

$$\alpha_{t+1}(i) = \max\{\alpha_t(i-1)\cdot a_{i-1,i}\cdot b_i(y_{t+1}), \alpha_t(i)\cdot a_{i,i}\cdot b_i(y_{i+1})\}$$

...(Equation 1)

$$\alpha_1(i)=b_1(y_1) \qquad (i=1)$$

$$\alpha_1(i)=0 \qquad (i\neq 1)$$

In the Equation (1), $a_{j,i}$ denotes a state transition probability from a state **j** to a state **i**, $b_i(yt)$ denotes an output probability with which a state **yt** is output in the state **i**, **yt** denotes the value of a feature order of a feature vector, and $\alpha_{t(i)}$ denotes a forward probability in the state **i** at a time **t**.

In this way, in the HMM speech recognition, the values of output probabilities for all states of state transition paths are required for each frame. In many cases, the output probabilities are given by a mixture multi-dimensional Gaussian distribution, which will be referred to as the mixture Gaussian HMM in this specification.

In the mixture Gaussian HMM, the output probability is given by a function of mixture multi-dimensional Gaussian distribution of Equation 2 which follows.

$$bs(y) = \sum_k \omega_k \prod_i \frac{1}{\sqrt{(2\pi\sigma_{ski})}} \exp\left\{-(y_i - \mu_{ski})^2 / \sigma_{ski}{}^2\right\}$$

....(Equation 2)

In the Equation (2) of mixture multi-dimensional Gaussian distribution, three mixture two-dimensional Gaussian distributions may be illustrate as in Fig. 7 for example. The three two-dimensional Gaussian distributions of Fig. 7 are expressed by an

Equation (3) which follows.

$$bs(y) = \omega_1 \left[\frac{1}{\sqrt{(2\sigma S_{11})}} \exp\left\{ -(y1 - \mu_{s11})^2 / \sigma_{s11}{}^2 \right\} \right.$$

$$\left. \times \frac{1}{\sqrt{(2\sigma S_{12})}} \exp\left\{ -(y2 - \mu_{s12})^2 / \sigma_{s12}{}^2 \right\} \right]$$

$$+ \omega_2 \left[\frac{1}{\sqrt{(2\sigma S_{21})}} \exp\left\{ -(y1 - \mu_{s21})^2 / \sigma_{s21}{}^2 \right\} \right.$$

$$\left. \times \frac{1}{\sqrt{(2\sigma S_{22})}} \exp\left\{ -(y2 - \mu_{s21})^2 / \sigma_{s21}{}^2 \right\} \right]$$

$$+ \omega_3 \left[\frac{1}{\sqrt{(2\sigma S_{31})}} \exp\left\{ -(y1 - \mu_{S31})^2 / \sigma_{S31}{}^2 \right\} \right.$$

$$\left. \times \frac{1}{\sqrt{(2\sigma S_{32})}} \exp\left\{ -(y2 - \mu_{s32})^2 / \sigma_{s32}{}^2 \right\} \right]$$

...(Equation 3)

Fig. 7 corresponds to a representation of the three two-dimensional Gaussian distributions, e.g., in a two-dimensional feature space y1, y2. In this case,

a mountain *A is represented by the first term of the Equation (3), a mountain *B is by the second term thereof, and a mountain *C is by the third term thereof. Fig. 8 is a view of the two-dimensional feature space when taken along a section 1 in Fig. 7 and viewed from its side. In the Equation (2), **k** denotes the number of mixture components or a mixture number, ωk denotes the height of a mountain, and a function given below is a one-dimensional normal distribution function for each dimension.

$$\frac{1}{\sqrt{(2\sigma_{ski})}}\exp\left\{-(y_i-\mu_{ski})^2/\sigma_{ski}{}^2\right\}$$

In this function, yi denotes a feature component of each dimension of a feature vector. In the Equation (2), the presence of a plurality of mountains is based on the fact that acoustic features differ among different speakers' ages or sexes even for the same word.

In order to speed up the computation of such mixture Gaussian distributions as shown by the Equations (2) and (3), it is effective to largely restrict a distribution to be computed and to convert part of the computation to a table. Further, for higher efficiency, it is often to also logarithmically evaluate a mixture multi-dimensional Gaussian distribution, but this is principally exactly the same

as in the integer processing. This will be explained with use of the high-speed technique of, e.g., the computation of the Equation (3).

In the viewpoint of speeding up the computation, it is possible to associate a feature vector with several standard patterns (vector quantization) and to define an output probability for each of the patterns as mentioned above.

The exemplary mixture Gaussian distribution of Fig. 7 will now be explained. In this example, for a feature vector present in a region 1, for example, the value defined by the Equation (3) is regarded as nearly equal to the value of the first term thereof (that is, the score of the second and third terms is nearly zero). Accordingly when it is only known that the feature is present in the region 1, the output probability of the Equation (3) can be acquired only with the computation of the first term (that is, the computation of the distribution *A).

In the case of the aforementioned processing, a feature space is divided into partial regions and the partial regions are linked to distributions to be computed. In this case, the vector quantization is often used for the linkage of the feature vector to the partial regions. The vector quantization is a method for considering a finite number of representative vectors in a feature space and for approximately represent a given point in the feature space in terms

of one of the representative vectors closest to the point. For example, when the feature space shown in Fig. 7 is represented by three points **a**, **b** and **c**, a feature vector in the region 1 corresponds to the point **a**.

Several efficient ones of such vector quantization methods are already suggested, but they basically select one of representative vectors closest in distance to a target point. For example, distances from the representative points such as **a**, **b** and **c** to the values of feature orders are computed, and one of the representative vectors having minimum one of the computed distances is selected. The vector quantization can be slightly smaller in computation amount than that when the mixture multi-dimensional Gaussian distribution is computed as it is, but its computation load is still large.

It is also possible to convert part of the computation of the output probability to a table to speed up the computation speed. Even in this case, the table can be made based on the vector quantization. However, when the vector quantization is carried out to link the output probability thereto, its quantization error becomes large and its recognition performance is deteriorated.

In order to avoid this, there can be employed a scalar quantization technique which resolves the computation into computations of the respective feature

dimensions, divides the respective feature dimensions into standard patterns and converts the respective computation results to tables. For example, a single Gaussian distribution shown by an Equation (4) which follows is transformed to a table.

$$\frac{1}{\sqrt{(2\sigma_{ski})}}\exp\left\{-(y_i-\mu_{ski})^2/\sigma_{ski}{}^2\right\}$$

...(Equation 4)

More specifically, a numeric value table having the value of yi linked to the value of the Equation (4) corresponding thereto is provided. This has basically the common principle, though the function is represented differently depending on whether or not to be a logarithmic system. In this case, its quantization error becomes small unlike the vector quantization.

As mentioned above, there are two types of the scalar quantization, that is, non-linear scalar quantization and linear scalar quantization. In the scalar quantization of a mixture Gaussian distribution, a function for each dimension is a single one-dimensional normal distribution, which is featured by being able to be defined when its average and dispersion are known.

In the non-linear scalar quantization, in order to reduce the number of numeric value tables, a

numeric value table relating to a one-dimensional Gaussian distribution of a representative average and dispersion is provided, parameter computation is carried out for various averages and dispersions to refer to the numeric value table on the basis of the parameter and feature component. However, this technique must necessarily perform the parameter computation for each feature component for the purpose of the table access. Further, even in the table reference, the access based on the parameter thus computed is not always a continuous array of access to the table, which results in that the address computation for the table reference requires multiplication and addition every time. This technique is described in the aforementioned literature "ON THE USE OF SCALAR QUANTIZATION FOR F AST HMM COMPUTATION", ICADDP 95, pp. 213-216. This technique also involves parameter computation requiring multiplication, subtraction, type conversion or shift operation for each feature component. Even for the table reference, access is carried out to an array of the parameter as an index. In this case, since it is not a continuous array of access, at the machine language (assembler) level, the computation of array address also require the computations of multiplication and addition (index times data length plus first address). Accordingly at the command level, two multiplications, two additions and subtractions, one type conversion or shift

operation, and two data loads (first address and numeric value data) are required.

Acquisition of the value of the numeric value table without carrying out the above computation can be realized, for example, by general linear quantization, which will be referred to as the linear scalar quantization, in this specification.

Shown in Fig. 9 is a relation between a numeric value table and a one-dimensional normal distribution when the linear scalar quantization is carried out. In the linear scalar quantization, a feature is quantized at intervals of a constant distance. For easy understanding of the quantization, when a distribution is divided into n-th power of 2 (2^n) of parts, the linear scalar quantization is equivalent to or synonymous with extraction of upper N bits of the feature component. The contents of this linear scalar quantization is shown in Fig. 10.

In the linear scalar quantization, since quantized representative points are fixed, the quantizing operation is required to be carried out once for each frame, that is, for each feature component. Further, since the representative point corresponds to an index as it is, a difference (which will be referred to as the offset, hereinafter) between the first and desired addresses in the numeric value table is (index times data length) and its computation is the same for the entire distribution, which means that the

computation is required to be carried out only once for each frame. And access to the necessary numeric value table can be calculated by a sum of the first address of each numeric value table and the offset. Thus the access is carried out eventually through one addition and two loads (first address and numeric value data).

In the computation (Equation (3)) of output probability of the mixture Gaussian HMM, it is important to reduce the amount of computation corresponding to a single Gaussian distribution (including a logarithmic type). Such computation for each feature component corresponds to a part of the output probability computation having a largest computation load, the number of computations is expressed by (the number of all models (the number of elements to be recognized times the number of states connected by left to right, 2N in the example of Fig. 6) times a mixture number times the number of feature dimensions). Thus a slight increase in the computation cost leads directly to an increase in the entire computation amount. Since this part of the linear scalar quantization produces entirely no computation other than the table access, this computation method is highly excellent from the viewpoint of a computation efficiency.

In the linear scalar quantization, however, since a numeric value table is required for each distribution with respect to a fixed representative

point, the number of numeric value tables or the amount of data becomes enormous, as mentioned above. Further, because of the speaker adaptive processing and noise adaptive processing, when a parameter (average or dispersion) of the mixture Gaussian distribution is modified, it correspondingly involves an enormous amount of computation. Even a modification in the numeric value table requires a great deal of processing.

In accordance with an embodiment of the present invention which will be explained in detail below, in the output probability computation using the mixture Gaussian distribution, part of the output probability computation is replaced by data table access of a one-dimensional normal distribution to realize a higher computational speed. At this time, a linear scalar quantization is provided which is characterized by employing an intermediate table or global table to enable high-speed computation of an output probability, whereby the amount of table data can be made less and the embodiment can flexibly cope with speaker adaptive processing, environmental (noise) adaptive processing, and so on.

<<Summary Of Speech Recognition System>>

Fig. 1 shows a block diagram of a speech recognition system in accordance with an embodiment of the present invention. The speech recognition system

shown in Fig. 1 non-exclusively include a speech recognition board 101, a microphone 107 and a monitor (display) 203. The speech recognition board 101 can be fully implemented on a one-chip LSI. The monitor 108, when a voice input device or the like is used for example, is not necessarily required.

The speech recognition board 101 has an A/D converter 102, a microprocessor (MPU) 103, a ROM (read only memory) 105 and a RAM (random access memory) 106. When the monitor 108 is added, it is required to additionally provide a video interface (VIF) 104.

The A/D converter 102 converts an analog voice signal inputted from the microphone 107 into a digital signal. The ROM 105, which is a read only memory, stores a program and data (such as a dictionary or HMM parameters) necessary for the speech recognition system therein. The RAM 106, which is a readable/writable memory, uses a work area or temporary area in the microprocessor 103.

Fig. 2 shows a detailed example of the MPU shown in Fig. 1. The microprocessor 103 is connected via a bus interface 118 to the ROM 105, RAM 106 and VIF 104. The operating program of the MPU 103 is sent via an instruction cache 110 to an instruction control unit 112 and decoded therein. The MPU 103 performs its computation control operation on the basis of its decoded result. Necessary data is sent via a data cache 117 from a load unit 114 to a register file 111,

or from the register file 111 via a store unit 115 to the data cache memory 117. Data stored in the register file 111 is as necessary processed by an integer unit 116 for integer computation, and processed by a floating point unit 117 for a floating point numeral. Its processed result is returned again to the register file 111 to be written in the memory via the store unit 115. In the data access, if the data cache 117 is hit, then no access to the external memory is carried out and reading from the data cache 117 is carried out or cache filling to the data cache is carried out. In the case of a cache miss, access to the external data memory is carried out and further a necessary entry is added to the data cache 117 from the external data memory. In the instruction access, if the instruction cache 110 is hit, then no access to the external memory is carried out and an instruction is fetched from the instruction cache 110. In the case of the cache miss, access to the external instruction memory is carried out and further a necessary entry is added to the instruction cache 110 from the external instruction memory.

Fig. 3 shows a summary of the entire processing procedure of the speech recognition system of Fig. 1 after a power is turned on to boot up the system until the power is turned off to stop the system.

In Fig. 3, a step 201 shows a start of the

operation. More concretely, this means the start of the operation of the system instructed by the turning on of the power supply (power on). When the system starts to operate, in a step 202, the system reads necessary data 250 from the ROM 105 and loads the necessary data 250 into the RAM 106 or data cache 117. In this case, when a high-speed nonvolatile memory is used with data seldom used or rewritten, such data is not positively loaded into the RAM 106 or the like but access is made directly to the ROM 105 to acquire data.

In steps 203 to 205, which form a sort of infinite or endless loop, the operation is repeated until an end instruction is executed. When the system judges the end of the operation in the step 205, the system terminates its operation (step 206). During the above loop operation, the adaptive processing (step 203) and recognizing operation (step 204) are executed as necessary.

The adaptive processing means the operation of modifying various parameters including HMM as necessary. Take the environmental adaptive processing for instance. In this case, noise is sampled in a noise environment used and the output probability of HMM is modified according to the sampled noise. In the mixture Gaussian HMM wherein its output probability is expressed by the aforementioned Equation (2), the modification means to modify an average and dispersion of each mixture Gaussian distribution. Data 252 is for

adaptation, while data 253 is for recognition.

The recognition operation (step 204) is executed as necessary with use of the HMM parameter (data 251) subjected to the above adaptive processing (step 203). In this example, the input voice data 253 from the microphone 107 is subjected to speech recognition and its recognized result 254 (such as text data) is output.

Shown in Fig. 4 is a summary of the aforementioned recognizing operation (step 204). When the recognizing operation is started in a step 211, a feature of the sample speech 253 is analyzed in a step 212 (feature analysis).

In the feature analysis, a speech waveform is divided by intervals of a fixed time (e.g., 10ms) and extracted into partial speech divisions (which will be referred to as frames) and the frames are analyzed in speech property on the assumption that the speech property will not vary (will be stationary). The speech property can be analyzed, for example, by frequency spectrum (computable by FFT) or by LPC coefficient (computable by an Levinson-Durbin recursive equation). These are generally represented by a group of parameters and thus are referred to as feature vectors. Through this feature analysis, the speech signal 253 is converted to a feature vector 255 for each frame. In this connection, an n-dimensional feature vector has **n** types of frequency components.

This series of feature vectors are referred to as an observation vector sequence.

In the next step 212, an output probability is computed. As has been explained in connection with Fig. 5, in the HMM, an output probability means a probability that each state outputs 'certain feature' speech. Accordingly, as explained in connection with the above Equation (2), the output probability is represented by a function of a feature vector indicative of the 'certain feature'.

There are two methods of the HMM speech recognition, that is, a method (discrete HMM) of quantizing a feature vector and providing an output probability as a probability function of the quantized vector and a method (continuous HMM) of providing an output probability as a probability function of the feature vector. In the present embodiment, the latter method is employed and the output probability is defined by a mixture Gaussian distribution.

In the case of the mixture Gaussian HMM, an output probability is given by the aforementioned Equation (2) for each HMM state as a feature vector function.

The computation of the output probability can be carried out concurrently with the recognition collation (Viterbi search) of a step 214. However, since its computation load is large, a necessary output probability is computed prior to the collation (search)

214 in order to avoid complex computation (step 213).

In the step 214, the score of each model is computed on the basis of the observation vector column obtained in the step 212 and the output probability 256 computed in the step 213. The word 'score' used herein can be defined, for example, as a (logarithmic) probability that a model given in Fig. 6 generates a pattern of a given feature vector column. A recognition candidate is set to be a model having a largest score. The score (which will be referred to as the Viterbi score, hereinafter) of the state transition sequence having the highest probability in each model is regarded as the score of the model to perform the Viterbi search.

<<Computation Of Output Probability Using Intermediate Table>>

Fig. 18 shows more details of the computing operation (step 213) of the output probability in the present embodiment.

In the present invention, in the (logarithmic) probability computation of the single Gaussian distribution, the feature component is equally divided into partial regions (linear scalar quantization), its corresponding computation result is previously converted to a numeric value table form, thus reducing its computation load. The benefit of the linear scalar quantization is that all mixture

distributions are quantized into an identical point with respect to each feature. That is, since the quantizing operation is shared by all the distributions, it is required only once for one frame. Further, when the index of the numeric value table is shared by the respective feature components, the offset (a difference between the first address of the table to be accessed and the address of the corresponding array element, which is generally computed by a product of the index and data length) of the numeric value table becomes also identical. Thus the offset finding operation of the numeric value table is also required only once for one frame. And, (unlike the non-linear scalar quantization), the operations necessary for the computation of the single Gaussian distribution require only addition (a sum of the first address of the array and offset thereof) and load store, whereby the computation can be realized in a time much shorter than that of the non-linear quantization.

However, such an approach, when the dispersion and average are modified due to adaptive processing or the like, requires modification of the numeric value table, (because the feature correspondence relation is fixed). To avoid such modification, an access pattern is controlled with use of an intermediate table having access addresses to the numeric value table set therein. Further, information for distribution selection and reduction is provided to

the intermediate table to simplify the computation, contents of which will be detailed below.

A step 1000 means to start the aforementioned step 213 of the output probability computation. In a step 1001, the feature vector (of any one of both integer and floating point types) analyzed in the step 212 is subjected to the linear scalar quantization, and an offset (which will be referred to as the feature offset or table offset, hereinafter) is computed for its linearly-scalar-quantized value (index). This computation can be easily carried out. For example, when the feature vector is of the integer type, the value subjected to the linear scalar quantization is divided by the entire number of quantizations and multiplied by a data length (the data length of one entire array) to compute the feature offset. As explained in Fig. 10, the linear quantization can be realized since, when a quantization range is divided into N-th power of 2 (2^N) of zones, upper N bits of the feature component can be obtained. Thus when the values of the quantization number and data length are converted to 2^N forms, the quantization can be implemented by one right shift. When this is expressed in the form of an equation for the feature vector is of the floating point type, the feature component is multiplied by a constant (definition zone length/quantization number times data length) to convert it to an integer type.

In the following description of the operation of Fig. 18, the aforementioned feature offset is used and the feature vector is not used in the computation. And the feature offset is expressed as data 1050.

In a step 1002, an access address of an intermediate table to be accessed for each distribution of each state is found from the feature offset found in the step 1001. The access address of the intermediate table is found by adding the first address (which varies from distribution to distribution as a matter of course) of the intermediate table defined for each distribution and the feature offset (which is the same for an identical feature dimension) together.

The intermediate table can be configured in any of a form in which the intermediate table 301 or 302 is arranged in a 1:1 relation to the one-dimensional Gaussian distribution as exemplified in Figs. 11 and 12 and a form in which the intermediate table 401 or 402 is extracted from the global table 400 commonly usable to a plurality of feature components. In the latter case, the global table 400 can be regarded as a set of many intermediate tables. In Figs. 11 and 12, exemplary intermediate tables are the intermediate tables 301 and 302. In Fig. 17, an exemplary global table is denoted by reference numeral 400. In Fig. 17, reference numerals 401 and 402 denote exemplary intermediate tables extracted from the global table 400.

In such a global table format as shown in Fig. 17, for example, first addresses of the intermediate tables 401 and 402 indicate the first locations (such as P1 and P2) of data areas to be extracted as intermediate tables from the global table 400. A technique for determining such a first location will be described in detail. As exemplified in Fig. 38, computation is carried out using the values of a table 410 having the values of averages and dispersions for respective feature components stored therein, or using an access pointer table 420 having their computed results previously stored therein. Pointers P0 to Pn of the access pointer table 420 denote the first locations of the intermediate tables 401 and 402 to be extracted for the respective feature components.

In the format of the intermediate tables 301 and 302 as shown in Figs. 11 and 12, on the other hand, the first addresses of the intermediate tables 301 and 302 mean the first addresses of the intermediate tables 301 and 302 respectively. The first addresses of intermediate tables to be defined for the respective feature components can be defined, for example, in the access pointer table 310 as the access pointers P0 to Pn as shown in Fig. 39.

The access pointer tables 310 and 420 are called an index table 1051 in Fig. 18. In Fig. 18, a table address 1055 corresponds to the value of an addition of the feature offset to the first address of

the intermediate table computed in the step 1002.

In this example, the intermediate tables 301 and 401 contain the addresses (offsets) of the numeric value table and information about distribution reduction. In the case of a normal distribution, as shown in Fig. 13, the numeric value becomes zero ($-\infty$ in a logarithmic type) as it goes away by a constant distance and more from an average (median) of the distribution. An uncorrelated multi-dimensional distribution is expressed by a product of one-dimensional normal distributions, so that, even for a single distribution, when it goes far away from its median, its numeric computation becomes meaningless. Accordingly, in such a numeric value data unnecessary zone, an area of the intermediate table corresponding to the unnecessary zone contains no addresses of the numeric value table and instead, for example, such distance data as defined by the following Equation (6) are stored therein.

$$d = -|(yi - \mu)|/\sigma \qquad \text{...Equation (6)}$$

The distance data of the Equation (6) have a negative value at all times. Further stored outside it is a value '0'. When the quantization number for the feature components is small, it is also possible not to store the value '0' as shown in Fig. 14.

The above distance data and value '0' are examples of the distribution reduction information. Fig. 15 shows an exemplary array of the above distribution reduction information to the single Gaussian distribution. In Fig. 15, a region E1 contains mapping addresses of data of the numeric value table, a region E2 contains the aforementioned distance information, and region E3 contains the above value '0'. It is natural that the region E2 or E3 may be absent depending on the distribution state of the one-dimensional Gaussian distribution based on the values of average and dispersion.

As shown in Fig. 16, with respect to the distribution reduction information, a distribution reduction condition 1 or 2 is judged. The value of the intermediate table accessed by the intermediate table 301 or 401 is judged. When the judged value is '0', the value of the multi-dimensional Gaussian distribution is regarded as '0' to interrupt the computation of the output probability relating to the multi-dimensional Gaussian distribution and to transfer the control to the operation of the next multi-dimensional Gaussian distribution. That is, the judgement of the distribution reduction condition 1 is whether or not the value of the intermediate table 301 or 401 is '0'. When the value of the accessed intermediate table 301 or 401 is negative, the value of the intermediate table 301 or 401 is regarded as

distance information and is multiplied with the distance information of the other component in the multi-dimensional distribution to find a product thereof. If the produce exceeds a fixed value, then the computation of the output probability of the multi-dimensional Gaussian distribution is interrupted and the control is shifted to the operation of the next multi-dimensional Gaussian distribution. Judgement of whether or not the accumulated value of the distance information exceeds the fixed value corresponds to the judgement of the distribution reduction condition 2. Only when the value of the intermediate table 301 or 401 is positive, the value of the intermediate table 301 or 401 is firstly regarded as the address of the numeric value table and the address data is fetched.

In the judgement of the distribution reduction condition 1 (step 1003) in Fig. 18, the value of the accessed intermediate table 301 or 401 is judged. When judging the value to be '0', the system interrupts the computation of the output probability of the multi-dimensional Gaussian distribution being processed and shifts the control to the operation of the next multi-dimensional Gaussian distribution (step 1011). When judging the value of the intermediate table 301 or 401 to be negative, the system regards the value of the intermediate table 301 or 401 to be negative and accumulates the distance information of the other components in the multi-dimension

distribution (step1004). Reference numeral 1056 means accumulated data on the memory. When the value of the accessed intermediate table 301 or 401 is positive, after completing the accumulating operation of the step 1004, the system judges the distribution reduction condition 2 to determine whether or not the accumulated distance value exceeds a predetermined value **a** (step 1005). When the accumulated distance value exceeds the predetermined value, the system interrupts the computation of the output probability of the multi-dimensional Gaussian distribution in question and moves to the operation of the next multi-dimensional Gaussian distribution (step 1011).

Only when the value of the intermediate table 301 or 401 is positive, the system regards the value of the intermediate table 301 or 401 as the address of the numeric value table and performs its corresponding operation. For example, in a step 1006, when a cache memory such as the data cache 117 is provided as shown in Fig. 2 and when data on the address is absent in the cache, the system prefetches data specified by the value of the intermediate table 301 or 401 into the data cache memory 117 from the numeric value table 1052 in the external memory. Such data prefetch is appropriately carried out when the data bus is unoccupied. This means that, when numeric value accumulation is later carried out using the value of the numeric value table, all or substantially all

necessary data 1053 are already stored in the data cache memory 117. In a step 1007, the system judges the presence or absence of the remaining single Gaussian components of the multi-dimensional Gaussian distribution being processed. In the case of the presence, the system returns to the step 1002 to compute the access address (addition) of the intermediate table relating to the single Gaussian distribution. At this time, it is unnecessary to again compute the table offset. This is because the feature component of the feature vector is already subjected to the linear scalar quantization as mentioned above.

In the operation of Fig. 18, the system accesses the intermediate table 301 or 401 for all the features in a first loop (steps 1002 to 1007). This can reduce the amount of wasteful computation during distribution reduction and also can produce no delay caused by prefetch (data prefetch of the numeric value table using the value of the intermediate table). For example, when determining in the judgement of the distribution reduction condition 1 that the numeric value is '0' during processing of one multi-dimensional Gaussian distribution, the system can interrupt the processing of the one multi-dimensional Gaussian distribution. Thus even when the system leads to such a circumstance, wasteful processing can be minimized.

It is also in principle possible to perform the operation of the step 1008 immediately after the

step 1006 without performing the branching operation of the step 1007. In this case, however, the prefetch function will not work effectively. (In general, it takes a little time in transferring data from the memory to the cache.) Further access to the numeric value table undesirably takes place even during the distribution reduction.

Therefore, in the present embodiment, the system access the numeric value table only for a distribution requiring computation to find a (logarithmic) value of the single Gaussian distribution in the step 1008. At this time, the numeric value data is present in the cache memory at all times and thus no cache miss penalty will take place.

The (logarithmic) value of the multi-dimensional Gaussian distribution is computed from the (logarithmic) value of the single Gaussian distribution. This computation is carried out as a product (a sum in the case of the logarithmic value) of the values of all the single Gaussian distributions. Accordingly in the step 1008, not only the system acquires the table value but also the system multiplies the acquired value by the already accumulated value (data 1057) (adds them together in the logarithmic type). In this case, when the system computes the first component, it requires '1' ('0' in the logarithmic type) as its initial accumulated value. The accumulated value is given as reference numeral

1057 in the drawing.

In a second loop (steps 1008 and 1009), when the operation of the step 1008 is executed for all components, its accumulated result becomes the value of the multi-dimensional Gaussian distribution. Accordingly in a step 1010, the system, in principle, saves the accumulated value stored in the register into the memory. Further, in the presence of a multi-dimensional Gaussian distribution not processed yet (step 1011), the system returns to the above step 1002. As in the above case, it is unnecessary to newly perform the computation of the table offset.

It goes without saying that the value of the multi-dimensional Gaussian distribution must be obtained by mixing the values of a plurality of distributions. Since the mixture is carried out by a sum (ADDLOG →addlog(a,b)=log{exp(a),exp(b)} in the logarithmic type) of all the values, the system performs the above operation together with the accumulated value and stores it in the register as a new accumulated value (step 1010).

In order to make a distinction between this accumulated value 1058 and the previous accumulated value 1057, the accumulated value of the data 1057 will be referred to as the multi-dimension accumulated data and the accumulated value of the data 1058 will be referred to as the accumulated mixture data, hereinafter. When the system computes the accumulated

mixture data 1058 with respect to all the single multi-dimensional Gaussian distributions, the system computes an output probability 256 in a step 1012. The accumulated mixture data fundamentally becomes the output probability 256, but it may be added by necessary constant data 1054 depending on the manner of the symbolic formula manipulations. (In the logarithmic type processing, the number of numeric value tables is reduced by separating parameters or the like.) In this case, necessary data may be extracted from the constant table 1054 to adjust the values. And the output probability 256 is eventually computed.

Through the operations shown in Fig. 18, the computation of one mixture Gaussian distribution has been completed. The above operations are executed for all the mixture distributions to be computed. (In the case of a general CMHMM, output probabilities are defined for all HMM states, in which case these all values must be found.) Accordingly, the effect of the simplified computation by Fig. 18 will be exerted on all the probability computations.

Figs. 19 and 20 show an example of the adaptive processing of the step 203 in Fig. 3. In Fig. 19, in the adaptive processing called so-called environmental adaptive processing, HMM parameters and more concretely the average and dispersion of a mixture Gaussian distribution are modified. Fig. 20 shows a processing procedure wherein the pointer of an

intermediate table is determined and updated for each one-dimensional Gaussian distribution from the modified dispersion and average of the Gaussian distribution.

Explanation will be made in detail as to the operations shown in Fig. 19. When the system starts its operation with a step 1101, the system analyzes a feature of noise data in a step 1102. This, when frequency spectrum is employed for example, can be executed by FFT (fast Fourier transform). In a step 1103, the system judges permission or non-permission of the adaptation on the basis of the analyzed data. The evaluation is carried out by comparing the noise property at the time of determining (modifying) the parameter and the current noise property.

The evaluation is considered to include various approaches, for example, to use the phase of a feature vector as the comparison reference and to evaluate correlativity of the frequency spectrum. When the correlativity is employed, a correlativity is found between the current noise spectrum (data 1150) and the spectrum (data 1151) at the time of determining the parameter and is used as an evaluated value 1152. This correlativity can be expressed, as an example, by an Equation (7) which follows.

$$\max_{k} = \left\{ \left(\sum_{i=1}^{N} si \right) \cdot N_{i-k} \right\} / (Ps \cdot Pn) \qquad \text{...(Equation 7)}$$

where

$Ps = \sqrt{\left\{\left(\sum_{i-1}^{N} Si\right)/N\right\}}, Pn = \sqrt{\left\{\left(\sum_{i-1}^{N} Si\right)/N\right\}}$, and N denotes the number of pieces of learning data.

Although the example wherein attention is focused on noise characteristic fluctuations has been given in Fig. 19, there is also a method wherein adaptation is forcibly carried out at intervals of a constant time. In this case, the step 1102 becomes unnecessary, the evaluated value 1152 contains time information (indicative of a time passed after the update) and the execution of the adaptive processing is judged after passage of a fixed time.

In either case, the adaptation is judged based on the evaluated value 1152.

When the system judges that the adaptation is necessary, the system performs the operations of steps 1105 to 1107. For example, if the noise feature vector is expressed by $n(\tau)=\{n1(\tau),n2(\tau),\ldots\}$ when $\tau=1, 2, 3, \ldots$, then the system modifies the average from the noise data, for example, as shown by an Equation (8) which follows.

$$\mu ki = \frac{\sum_{\tau=1}^{T} Kk(\tau)\cdot ni(\tau)}{\sum_{\tau=1}^{T} Kk(\tau)} \qquad \text{...(Equation 8)}$$

where

$$Kk(\tau)=\frac{\omega k\prod\left[\left\{(1/2\pi\sigma i)^{1/2}\right\}\exp\left\{(ni-\mu i)^{2}/\sigma i^{2}\right\}\right]}{\sum\omega s\prod\left[\left\{(1/2\pi\sigma i)^{1/2}\right\}\exp\left\{(ni-\mu i)^{3}/\sigma i^{2}\right\}\right]}$$

, **n** is learning data.

Similarly, the system modifies the dispersion in a step 1106, for example, as shown an Equation (9) which follows.

$$\sigma k=\frac{\sum_{\tau=1}^{T}Kk(\tau)\cdot ni^{2}(\tau)}{\sum_{\tau=1}^{T}Kk(\tau)}-\mu ki^{2} \qquad \text{...(Equation 9)}$$

where

$$Kk(\tau)=\frac{\omega k\prod\left[\left\{(1/2\pi\sigma i)^{1/2}\right\}\exp\left\{(ni-\mu i)^{2}/\sigma i^{2}\right\}\right]}{\sum\omega s\prod\left[\left\{(1/2\pi\sigma i)^{1/2}\right\}\exp\left\{(ni-\mu i)^{3}/\sigma i^{2}\right\}\right]}$$

In a step 1107, the system also modifies a mixture component weight, for example, a shown by an Equation (10).

$$\omega k=\frac{1}{T}\sum_{\tau=1}^{T}k(\tau) \qquad \text{...(Equation 10)}$$

where,

$$Kk(\tau) = \frac{\omega k \prod \left[\left\{ (1/2\pi\sigma i)^{1/2} \right\} \exp \left\{ (ni - \mu i)^2 / \sigma i^2 \right\} \right]}{\sum \omega s \prod \left[\left\{ (1/2\pi\sigma i)^{1/2} \right\} \exp \left\{ (ni - \mu i)^3 / \sigma i^2 \right\} \right]}$$

The analysis of the step 1102 does not have to be necessarily the feature analysis used in the speech recognition. However, , the feature of the steps 1105 to 1107 is the feature analysis parameter used in the speech recognition as a matter of course. Accordingly if the analysis of the step 1102 is not the feature analysis used in the speech recognition (if the speech recognition is LPC cepstrum and the operation of the step 1102 is the frequency spectrum as an example), then the system executes necessary operations prior to the operations of the steps 1105 to 1107.

The operations of the steps 1105 to 1107 are carried out for all the mixture distributions (step 1108). And after the modification for all the mixture distributions, the system contains analyzed data 1150 of the noise in an assumption characteristic 1151 (step 1109), and terminates its operation at a step 1110.

<<Global Intermediate Table>>

Through the operations of Fig. 19, the average and dispersion of the one-dimensional Gaussian distribution in the mixture distribution are modified. This manner is exemplified in Figs. 11 and 12. In this manner, when the average and dispersion of the one-dimensional Gaussian distribution are modified, the

manner of access to the intermediate tables 301 and 302 as shown in Figs. 11 and 12 is modified so that suitable access to the numeric value table can be realized without rewriting the numeric value table while such linear scalar quantization is carried out as shown in Figs. 9 and 10.

The insertion of the intermediate table 301 causes generation of an additional table access. However, as explained in the operation of Fig. 18, when the address of the numeric value table is contained in the intermediate table 301 and loop division and prefetch are carried out, even insertion of the intermediate table 301 upstream of the numeric value table can cause the amount of operation increased by the access to the intermediate table 301 to be suppressed to a low level. This has been already explained in connection with Fig. 18.

Note now that, when the dispersion and average are modified through the operation of Fig. 19, how it reflects on the intermediate table. For example, if the address of the numeric value table to be contained in the intermediate table is rewritten, the contents of the intermediate table 301 can be rewritten into contents of the intermediate table 302 so that access from Fig. 11 to Fig. 12 can be made according to a change in the dispersion and average. The rewriting from Fig. 11 to Fig. 12 means that the intermediate table 301 shown in Figs. 11 and 12 must

be, in principle, defined for all the one-dimensional Gaussian distributions. However, if the intermediate table 301 or 302 is given for each one-dimensional Gaussian distribution, then it alone results in an enormous amount of data and a table updating cost involved by the modification of the average and dispersion is similarly enormous.

In order to avoid such a problem, such a global table 400 (which will also be referred to as the global intermediate table) as shown in Fig. 17 is provided by only one. Shown in the drawing is a basic structure of the global intermediate table 400. In Fig. 17, white array elements denote addresses (positive value) of the numeric value table, black array elements (negative value) contain the distance information (negative value), and the other elements contain a value '0'. The number of data regions in the X-direction arrays is set to be larger than the quantization number of the feature component. This is because the first location of the intermediate table is shifted to the X direction according to the average value of the one-dimensional Gaussian distribution and thus it is required to take an extra data region in the X direction.

When an average (μ) is the average ($\mu 0$) of a standard table, the above global intermediate table 400 contains addresses (offsets) of the numeric value tables having various dispersions as well as the

aforementioned distance information. The example of Fig. 17 shows a pattern when the left-side column has a largest dispersion and, as it goes to the right, the dispersion is decreased.

When such a global intermediate table 400 is prepared, a pattern of the intermediate table corresponding to a given average and dispersion can be surely provided on the global table 400. That is, the position of the global intermediate table 400 in the horizontal direction (Y direction) is determined by the dispersion (σ) of the target one-dimensional Gaussian distribution. The array of the column selected by this dispersion is the array of address data for access of the numeric value data realizing the one-dimensional Gaussian distribution having the average (μ) in its middle. With respect to the desired average (μ), the access start position to the array data of the column determined by the dispersion (σ) is shifted to the vertical direction (X direction) according to the average. In other words, the array data of the column determined by the dispersion (σ) is shifted in the vertical direction.

In Fig. 17, for example, when the dispersion is expressed by σ and the average is by $\mu 0$, the pattern of the intermediate table 401 corresponding to a distribution 1 is expressed by array elements having the first address in Fig. 17. Similarly, a distribution 2 having a dispersion σ' and an average

μ' is expressed by the intermediate table 402 of array elements having the first address P2 in Fig. 17. The first address (which will be referred to merely as the access pointer, hereinafter) P1 or P2 of the intermediate table 401 or 402 corresponding to the distribution may be previously transformed to such a pointer table 420 as shown in Fig. 38. The pointer table 420 forms part of the HMM data. In the address computation 1002 of the feature components in the operation of Fig. 18, the processing sequence of the feature components can be previously determined. Thus it is only required to previously convert the first addresses of the intermediate tables to a table, that is, to prepare the table in such a manner that a necessary one-dimensional Gaussian distribution can be identified according to the sequence. Such a table is, for example, the pointer table 420 shown in Fig. 38. By extracting from the pointer table 420 the first address of the intermediate table to be added to the feature offset computed in the step 1001 in Fig. 18, the necessary intermediate table can be extracted from the global table 400.

By using the pointer table 420, the global intermediate table 400 can be used as a reference-exclusive table (which can eliminate the need for rewriting the contents of the table). Thus even use (shared use) of the global intermediate table 400 as overlapped with the operation of the other Gaussian

distributions will not cause any problems at all. And when the access pointer (P1 for the distribution 1 or P2 for the distribution 2) defined on the pointer table 420 is regarded as the first address of the intermediate table, the system can perform the operation as if an intermediate table were present as an entity. Even when the global intermediate table 400 is used, the operation of Fig. 18 will not be changed at all.

For the purpose of coping with the modification of the average and dispersion of Fig. 19, the need for rewriting the intermediate table itself can be completely removed, and it is suffice to merely compute an access pointer corresponding to the average and dispersion and reflect it on the access pointer table 420. That is, when the dispersion and average were changed through the adaptive processing, the system can cope with it without rewriting the intermediate table by changing the first address (the value of the access pointer) of the original intermediate table according to a change in the dispersion and average. For example, assume that the pattern of the intermediate table of the distribution 1 corresponding to the before-adaptation is expressed by the array elements having the first address P1 in Fig. 17. Then when the pattern of the intermediate table of the distribution 1 corresponding to the after-adaptation is changed to the array elements having the

first address P2 in Fig. 17, it is only required to change the point (access pointer) of the first address of the intermediate table of the distribution 1 from P1 to P2. This operation may be carried out for such a pointer table 420 as exemplified in Fig. 38.

In the summary of the operation for the above, the system first selects a column (dispersion column) having a dispersion closest to the modified dispersion and for the modification of the average, moves the first location of the column vertically on the basis of a difference between the average of the standard Gaussian distribution and the modified average.

Explanation will be further made in more detail as to the modification of the first address of the intermediate table to be extracted. Consider first the operation of enabling access to the one-dimensional Gaussian distributions having various dispersions and averages using the standard table.

Consider a case where, when $f_0(x_0)=\exp\{-(x_0-\mu_0)/\sigma_0\}$, the system computes the value of $f(x)=\exp\{-(x-\mu)/\sigma\}$ having a given average and dispersion with use of a standard table $x_0 \Rightarrow f_0(x_0)$. Then x_0 satisfying a relation of $f_0(x_0)=f(x)$ is expressed using **x** as follows. From $f_0(x_0)=f(x)$, equation reduction is done as follows.

$$\log\{f_0(x_0)\} = \log\{f(x)\}$$

$$\log\{\exp\{-(x_0 - \mu_0)/\sigma_0\}\} = \log\{\exp\{-(x - \mu)/\sigma\}\}$$

$$(x_0 - \mu_0)/\sigma_0 = (x - \mu)/\sigma$$

$$\therefore x_0 = (\sigma_0/\sigma)(x - \mu) + \mu_0$$

This equation means that a value $(x_0 - \mu)$ at a location x_0 at the time of setting an average location to an origin is equal to a value determined by a value $(x - \mu)$ at a location **x** at the time of setting the average location as the origin and by a value σ_0/σ. Further reduction of the above equation results in:

$$x_0 = (\sigma_0/\sigma)(x - \mu + \mu_0\sigma/\sigma_0)$$

where, when $\alpha = \sigma_0/\sigma$ and $\beta = \mu - \mu_0\sigma/\sigma_0$, the above equation can be expressed as $x_0 = \alpha(x - \beta)$.

Consider next, when $C(x) = a(x - \beta)$ (where α and β are the same as those in the above case), how to obtain a value of C(x) having a given average and dispersion using a simple table. C(x) should be essentially considered as a three-dimensional table (x, α, β). However, a two-dimensional table defining $x_0 = \alpha$ x is assumed and at the access time, a location is shifted in the X direction by $-\beta$ to obtain C(x) as exemplified in Fig. 37. The first address of the intermediate table after the adaptation is determined on the basis of the first location of the table eventually obtained through the shift $-\beta$. In Fig. 17, the first address becomes P2 of the distribution 2,

that is, the modified value of the corresponding intermediate table pointer.

Fig. 20 shows an example of a general processing procedure of determining the value of the corresponding access pointer for the dispersion and average of a Gaussian distribution modified by the adaptive processing of Fig. 19. When the system starts its operation, computes the values of α and β with use of a standard average/dispersion value 1251 and new average value 1153 and dispersion value 1154 obtained through the adaptation (step 1202). And as mentioned above, the table line (column) of the global intermediate table 400 is determined on the basis of the value α (step 1203). Further, the table first location is determined with use of the value β (step 1204). An address is computed from the determined table line and table first value (step 1205). In the computation, data (header of an index table) 1253 having a table structure is referred to. For example, assuming that the location of the table line is denoted by T, the first location is by S, the number of table elements in one line is by E, one element has a data length of 4 bytes, the first address of the global intermediate table is by A0 and the address is of a byte type; then the address of a two-dimensional array is computed by the following equation.

$A=A0+4\cdot\{(T-1)\cdot E+S-1\}$

'A' corresponds to the value of the access pointer

after the adaptation.

The operations of the steps 1202 to 1205 are repeated for all the distributions. Through the repetitive operations, the first address of the intermediate table used in the operation of Fig. 18 is associated with the global intermediate table 400 of Fig. 17 as its address.

As will be clear from the foregoing explanation, the global intermediate table 400 can be referred to on the basis of the values of the average (μ) and dispersion (σ), but the foregoing description has been made using the pointer table 420 containing the pointer (access pointer) of the first address of the intermediate table to be extracted. In this case, as exemplified in Fig. 38, the feature components are provided with the access pointers P0 to Pn respectively. It will be obvious from the above explanation that the value of the access pointer can be computed on the basis of the dispersion and average. Accordingly the access pointer can be uniquely associated with the dispersion and average of the corresponding distribution. And as shown in Fig. 38, the table 410 having the dispersions and averages of the feature components may be prepared and the values of the access pointers P0 to Pn may be computed and found every time on the basis of the table. In the case of using the table 410, however, the amount of operation at the time of the adaptation is decreased,

while the amount of computational operation necessary for reference to the global intermediate table 400 is increased. Conversely, in the case of an arrangement using the pointer table 420, the amount of computational operation for reference to the intermediate table is decreased, while the amount of operation necessary for the adaptation is increased. The average and dispersion of each feature component or the access pointer of each feature component is held in the system as HMM data (251 in Fig. 3) together with a state transition probability necessary for computation of the mixture HMM and so on.

Figs. 42 to 44 show an example of the numeric value table of a one-dimensional Gaussian distribution. In Fig. 42, the value shown by the above Equation (4), that is the value of an expression surrounded by a rectangle R1 in Fig. 42 is provided for each desired dispersion. The value possessed by the numeric value table is set to be in a range of -4σ to 4σ. The value range is associated with the structure of the intermediate table in Fig. 13 for the purpose of the distribution reduction. The data structure of the numeric value table has properties common with the intermediate table, that is, numeric value data relating to the dispersion assumed by the intermediate table. When such numeric value data is employed, data referred to by the numeric value table must be summed up. Thus from the viewpoint of computational digit

number or computational accuracy, it is desirable that the microprocessor 103 for performing computing operation of the mixture HMM be provided with such a floating point unit as shown in Fig. 2.

The numeric value table shown in Fig. 43 contains logarithmic data values which can be used even for the integer processing. In this case, the value of an expression surround by a rectangle R2 is contained in the numeric value table of Fig. 42. Further, even a logarithmic value of mixture component weight surrounded by a rectangle R3 must be held in the table. A big difference of Fig. 43 from Fig. 42 is that Fig. 43 can cope with even the integer processing.

Figs. 40 and 41 collectively show a table access technique for the aforementioned probability computation using the multi-dimensional Gaussian distribution.

In Fig. 40, the HMM data contains, for example, the values of access pointers for the respective feature components as the pointer table 420. For example, the value of a feature component for one feature component is P1. The value P1 is changed to P2 by the adaptation. In this computation, the access pointer value P2 is determined on the basis of the dispersion and average uniquely determined by P1 and the dispersion and average changed by the adaptation. A feature offset is computed by the feature extraction for each feature component, and further the value P2 of

the access pointer to be added thereto is read to thereby compute the reference address of the intermediate table. When the global intermediate table 400 is read based on the reference address, the value of the one-dimensional Gaussian distribution relating to the predetermined dispersion and average of the feature component is read out from the numeric value table on the basis of the read address.

As will be obvious from the aforementioned description, the acquisition of the one-dimensional Gaussian distribution corresponding to the feature component in the mixture HMM computation in the speech recognition mode can be realized through simple operation such as addition of the feature offset and access pointer while requiring no complex parameter computation. At the time of the adaptation, it is only required to modify the access pointer while completely eliminating the need for modifying the global intermediate table 400 and numeric value table 1052.

In Fig. 41, prior to the computation of the output probability, the feature offsets are required to be previously found for the feature components of the feature vector. And the global intermediate table 400 is accessed on the basis of the access pointer values and feature offsets for the respective feature components to sequentially acquire the addresses of the numeric value data of the one-dimensional Gaussian distribution. And when all the addresses of the

numeric value data of the one-dimensional Gaussian distribution included in one multi-dimensional Gaussian distribution are acquired, the numeric value data are accessed based on the acquired addresses. At this time, if data prefetch to the address for the numeric value data access is already done by then, then substantially no cache miss will take place at the time of accessing the numeric value table. The prefetch can be carried out as necessary at timing when the MPU 103 does not perform data access. Therefore, even when access to the global intermediate table 400 is made prior to access to the numeric value table, the acquisition of the numeric value data will not be delayed. Further, so long as the global intermediate table 400 is previously stored in a high-speed RAM 106 or the like having the microprocessor 103 built therein, the access time to the global intermediate table 400 can be made as small as substantially negligible. When it is desired to modify the dispersion and average due to the adaptation, it is only required to modify the value of the access pointer pointing the first location of the intermediate table to be extracted, as mentioned above.

<<Portable Information Terminal Device>>

Shown in Fig. 21 is an exemplary outside view of the portable information terminal device 120 to which the speech recognition system is applied. Shown

in Fig. 22 is a block diagram of the portable information terminal device 120. The illustrated portable information terminal device 120 includes the aforementioned speech recognition function, the functions of a small computer device and a portable telephone function, though not specifically limited to the specific example. A display 108 and keyboard 123 are arranged in the central part of a casing; and microphones 107, 1301 and loudspeakers 1307, 1308 at the ends of the casing.

In Fig. 22, an MPU 103, ROM 105, RAM 106, VIF 104 and display 108 are the same as those in the circuit of the speech recognition system already explained in Fig. 1, and are used commonly to the aforementioned speech recognition function, the function of the small computer device and the portable telephone function.

In Fig. 22, a portable telephone unit (PHS) is denoted by reference numeral 1303. The portable telephone unit 1303 can talk with another portable telephone unit or general wired telephone via an antenna 1309. The loudspeakers 1307 and 1308 are connected to the MPU 103 via the D/A converters 1305 and 1306 respectively. A peripheral circuit 1302 realizes or implements an infrared interface circuit, flash memory card interface or the like.

The portable information terminal device 120 is assumed to be of a two-channel microphone input

type, though not limited to the specific example. The microphone 1301 can be connected to the MPU 103 or PHS 1303 via an A/D converter 1204. The microphone 107 can be connected to the MPU 103 via the A/D converter 102. The both microphones 107 and 1301 are used for speech recognition or telephone, an application configuration of which will be detailed later.

The portable information terminal device 120 uses the battery 121 as an operating power source from the viewpoint of importance in its portability. In order to prolong the operating time of the portable device on the battery 121, its requirement of low power consumption is strictly demanded when compared to the system using a commercial power supply its operating power source at all times. In order to meet the demand, an MPU having relatively small operational speed (operational clock frequency), MIPS (million instruction per second) value or power consumption tends to be employed as the MPU 103. For example, an MPU having a power consumption of about 1W, an operational clock frequency of about 200MHz and a data processing ability of about 300MIPS can be employed as the MPU 103.

At this time, when the aforementioned speech recognizing operation is carried out using the MPU 103, the linear quantization technique and global intermediate table technique are employed for the computation of the mixture multi-dimensional Gaussian

distribution, so that the high-speed computational operation in the speech recognizing operation as well as the high-speed parameter change at the time of the adaptation can be realized. Thus even when such an MPU 103 having a relatively low data processing ability is employed, speech recognition can be carried out at a speed as high as sufficiently practical without impairing the real-time and quick performance of the speech recognition.

A program for speech recognition control based on the linear quantization technique and global intermediate table technique for the computation of the mixture multi-dimensional Gaussian distribution is stored, for example, in the ROM 105. This ROM is a recording medium readable by the MPU 103 as a computer. When the ROM 105 is an electrically-rewritable, non-volatile memory such as a flash memory, the speech recognition control program can also be externally loaded into the ROM for its execution. For example, the necessary speech recognition program can be transmitted to the ROM from a CD-ROM drive (not shown) interfaced with the peripheral circuit 1302. At this time, the CD-ROM drive is given as an example of the computer-readable recording medium having the speech recognition control program stored therein.

<<Two-Microphone Type Noise Adaptive processing>>

There is a known technique (such as ANC

(adaptive noise canceller)) for using two microphones to cancel a noise component from a speech to be recognized. Explanation will be made as to a case where the above technique is employed to perform the noise adaptive processing with use of two microphones. The microphone 107 can be used as a main microphone to pick up a speech together with noise. The other microphone 1301 can be used as a sub-microphone to pick up a relatively large noise component when compared with a signal component. For example, this is realized by selecting the directivity and array of the both microphones 107 and 1301.

Fig. 34 shows the principle of two-microphone type noise adaptive processing. In a speech duration, noise and speech are overlapped with each other and sampled by the main microphone 107. The sub-microphone 1301 samples noise exclusively and its sampled noise signal contains substantially no speech signal component. The feature of the noise included in the signal obtained through the main microphone 107 is different from the feature of the noise obtained through the sub-microphone 1301, as a matter of course. Thus in a speechless duration, the characteristics of the main and sub-microphones 107 and 1301 are evaluated. Assuming for example that the characteristic of the main microphone 107 is denoted by $fm(\omega)$ and the characteristic of the sub-microphone 1301 is by $fs(\omega)$, then it can be expressed as $fm(\omega)=$

$\alpha(\omega)\cdot fs(\omega)$ due to distortion having a multiplication property. In the speechless duration, the above $\alpha(\omega)$ can be determined on the basis of the signals from the main and sub-microphones 107 and 1301. In a speech duration where an input from the main microphone 107 exceeds a predetermined threshold value, an input from the sub-microphone 1301 is subjected to noise analysis to compute $fs(\omega)$. And the characteristic of $fm(\omega)$ is corrected based on $\alpha(\omega)\cdot fs(\omega)$. Thereafter, the average, dispersion and mixture component weight shown in Fig. 19 are corrected and further the value of the access pointer of the pointer table 420 is modified as explained in Fig. 20.

Fig. 23 shows, in detail, an example of a processing procedure when the noise adaptive processing is carried out using two microphones in the portable information terminal device 120.

When the system boots up in the step 202 and reads system data from the necessary data 250, the system judges in a step 1401 whether or not a speech was input to the main microphone 107. When determining no speech input in a step 1402, system returns to the operation of the step 1401 via a step 1403. This operation, which forms a sort of endless loop, is repeated until a speech is input to the main microphone.

In the step 1403, the characteristic of the main microphone 107 is compared with the characteristic

of the sub-microphone 1301 for evaluation. This is because a difference in characteristic between the main microphone and sub-microphone is previously corrected in order to estimate the characteristic of noise from the main microphone from the noise of the sub-microphone.

When determining in the step 1402 the speech input to the main microphone, the system analyzes in a step 1404 the feature of the speech data (data 1451) of the sub-microphone by the sub-microphone noise analysis (step 1404). And using a characteristic 1452 of the main and sub-microphones evaluated in the step 1403, the system corrects the analyzed result obtained in the step 1404 (step 1405). And on the basis of the analyzed result in the step 1404, the system judges in a step 1406 whether or not to perform adaptation. When determining the performance of the adaptation, the system performs the noise adaptive processing using a result corrected in the step 1405 (step 1407). The operation of the step 1407 can be implemented by substantially the same technique (a difference from Fig. 19 is that the need for performing operations associated with judgement of adaptation or non-adaptation can be eliminated) as that in Fig. 19. In this example, the system updates the pointer table 420 of the access pointer pointing the first address of the intermediate table on the basis of data 1453 of the modified HMM parameters (average and dispersion of the

mixture Gaussian distribution)(step 1408). This updating can be carried out, for example, by the technique of Fig. 20. The updated pointer table 420 is thereafter used for output probability computation 212 or Viterbi search 214.

With respect to the two-microphone type speech recognition, as another example of the aforementioned ANC technique, there can be applied a known technique (such as a beam-former) wherein speech information obtained from a pair of stereo microphones is separated into signal-component-emphasized data and noise-component-emphasized data, and the ANC technique is applied thereto.

<<Speech Recognition In Transceiver Type Speech>>

In the portable information terminal device 120 shown in Figs. 21 and 22, objects to be subjected to the speech recognition include two types, that is, a speech (caller speech) from a speech caller of the portable telephone unit 1303 and input speech (terminal speech) from the main microphone 107 of the terminal device 120. With respect to the speech recognition of the above caller speech (caller speech recognition) and the speech recognition of the terminal speech (terminal speech recognition), the speech recognition of the transceiver speech is firstly considered. That is, as shown in Fig. 35, one of both speeches is exclusively recognized by switching between the caller speech and

terminal speech. Such switching operation can be implemented by means of the switch 1302SW which switches between the speech input from the terminal and the received caller speech. In Fig. 22, the switch 1302SW is illustrated as a circuit included in the peripheral circuit 1302 for simplicity. The features of the both speeches are estimated to be considerably different from each other. At this time, when different HMM numeric value tables are separately provided for the caller speech and terminal speech, the amount of its data becomes too large; whereas, when a common HMM numeric value table is provided for the both speeches, the amount of operations necessary for the adaptation whenever change-over is made between the caller speech and terminal speech becomes enormous and thus it is estimated that real-time processing can be fully disabled. The HMM numeric value table and global intermediate table are used commonly to the caller speech and terminal speech, and the pointer table 420 is separately prepared for each of the caller speech recognition and terminal speech recognition. And the separately-prepared pointer tables are selectively used for each of the inputs. In the caller speech recognition, a pointer table allocated thereto is used to access the global intermediate table. In the terminal speech recognition, a pointer table allocated thereto is used to access the global intermediate table. In Fig. 40, reference symbol 420-2 denotes a

caller pointer table and symbol 420-1 denotes a terminal pointer table.

Shown in Fig. 24 is an example of a processing procedure of speech recognition in the transceiver type speech using the portable information terminal device 120.

When the system starts its operation in the step 201, the system reads out system data from the ROM 250 in the step 202. In this example, the system judges in a step 1501 whether the speech is from the caller or from the terminal by utilizing the feature that the speech from the terminal and the speech from the caller can be input independently of each other. For example, the system judges it on the basis of the state of the switch 1302SW for change-over between the caller speech and terminal speech. When determining the speech input from the terminal, the system inputs the terminal speech data as an object to be subjected to the speech recognition in a step 1503. When determining the input received from the caller, the system inputs the call speech data from the sub-microphone 1301 as an object to be subjected to the speech recognition in a step 1504. In a step 1505, the system extracts a speechless duration from each input and analyzes its noise property. In a step 1406, the system judges whether or not to perform adaptation on the basis of data of the speechless duration of the input speech. When determining the adaptation, the

system modifies HMM parameters such as dispersion and average in the adaptive processing of a step 1407, and correspondingly updates the value of the pointer of the pointer table 420 in the step 1408. The subsequent operations are exactly the same as those in Fig. 23 and thus detailed explanation thereof is omitted.

<<Speech Recognition In Separate Type Speech>>

As a speech recognition technique for the caller and terminal channels using the portable information terminal device 120 exemplified in Figs. 21 and 22, there is secondly considered a speech recognition in the separate type speech. That is, as shown in Fig. 36, the caller speech (receiver speech) and the terminal speech (sender speech) are provided as mixed to enable the speech recognition. In this example, the switch 1302SW becomes unnecessary. Even in this case, the situation is similar to the above, that is, the HMM numeric value table and global intermediate table are used commonly to the caller speech and terminal speech and the pointer table of the intermediate table is separately prepared for each of the caller and terminal speech recognitions. However, the speech durations of the terminal and caller channels must be detected separately. As a result, even when the caller speech is overlapped with the terminal speech, the system can cope with it. In this connection, when the intermediate table is allocated to

each feature component without using the global intermediate table, the intermediate table must be provided separately for each of the caller and terminal channels.

Fig. 25 shows an example of a processing procedure of speech recognition when the portable information terminal device 120 is used for the separate type speech. In this example, the system is configured which has two sets of parameters adapted and adjusted to the caller and terminal channels. In this case, the numeric value table 1052 and global intermediate table 400 are the same for each of the caller and terminal channels, and thus it is only required to have only two sets of the pointer tables 420 having the access pointer of the intermediate table.

In Fig. 25, when the system starts its operation in the step 201, the system first boots up in the step 202. The present system, utilizing that the terminal speech input is provided as separated from the caller speech input, performs the operation for each of the both channels. In the step 1503, the system inputs a speech from the terminal channel. If adaptation is necessary, then the system detects the speechless duration in a step 1505-1 and performs the noise adaptive processing in a step 1407-1. And in response to the adaptation, the system updates the terminal pointer table 420-1 of the intermediate table in the

step 1408.

Operations similar to the above are carried out even for the caller channel. As in the present device, the device is provided as integral with the portable telephone unit 1303, the system input a speech signal to be recognized from the caller channel in the step 1504. Thereafter, the system perform the operations of steps 1505-2, 1407-2 and 1408-2.

It should be noted that the speech input system and the pointer table of the intermediate table separately require two channels, but the speech recognition control program and global intermediate table be sufficient to be each single (commonly used). Although the system does not perform recognizing operation separately for the terminal and caller channels but can have a performance and function equivalent to those when the system performs the recognizing operation for the channels separately.

In a step 1601, the system performs overlap adjustment. This adjustment is carried out when the terminal and caller channel speeches are overlapped (for example, when the caller and receiver talk at the same time). This also can be realized, as a simple example, by detecting the speech durations for the input speeches, waiting the end of the previous-detected speech duration and thereafter processing the later-detected speech duration.

In this way, after having acquired a signal

(having attribute data or flag for distinction between the terminal and caller channels) of the speech duration, the system performs feature analysis in the step 212, computes the output probability in the step 213 and performs Viterbi search in the step 214 to thereby obtain a recognition result (data 254-2) having a channel attribute. The 'channel attribute' means attribute data for distinction between the terminal and caller channels.

In the above operations, even for such operation as requires a plurality of channels of data sets, it is only required to have the pointer table 420 of the intermediate table alone for each channel. In other words, only the pointer table of the intermediate table is provided one for each of the two channels, and the global intermediate table 400 and numeric value table 1052 are commonly used for all the terminal and caller channels.

<<Speech Recognition Supporting Speaker Adaptive processing>>

Fig. 26 shows an example of a procedure of the speech recognizing operation in the speech recognition system which performs speaker adaptive processing and noise adaptive processing. In this case, the system performs the adaptive processing at intervals of a fixed time on the basis of time information 1752.

As in the previous embodiment, when the system starts its operation in the step 201, the system first boots up in the step 202. After the system boots, the system inputs speech data in a step 1701, and increments the time information 1752 in a step 1702. In this connection, the time information may be of a clock unit or frame unit. In the judgement of performance or non-performance of adaptation (steps 1703-1 and 1703-2), the system judges whether or not the time information 1752 is not smaller than a fixed value. If the time information is not smaller than the fixed value, then the system executes the adaptation. If the system does not execute the adaptation, the system moves to the step 212 to start the speech recognition.

When performing the noise adaptive processing, the system first inputs noise data in a step 1704-1 and correspondingly modifies parameters in a step 1705-1. For example, in the two-microphone type, the above operations may be the same as those of the method (steps 1404 to 1407) of Fig. 23. And in a step 1706-1, the system modifies the access pointer table 420 of the global intermediate table according to the modified dispersion/average 1453 and resets the time information 1752 (e.g., sets it to 0). And the system performs speech recognizing operation (in the step 212 to 214).

The same holds true even for the speaker

adaptive processing. As in the noise adaptive processing, in the adaptation judgement of a step 1703-2, the system executes the adaptation when the time information 1752 is not smaller than a fixed value. However, it is not necessarily required that the time interval of the speaker adaptive processing be the same as the time interval of the noise adaptive processing. In a step 1704-2, unlike the noise adaptive processing, the system extracts a speech duration. In a step 1705-2, the system performs so-called 'speaker adaptive processing without teacher'. The system, on the basis of the modification, updates the pointer table 420. The 'speaker adaptive processing without teacher' means a speaker adaptive processing system which does not perform previous leaning for adaptation.

The above noise adaptive processing and speaker adaptive processing take place like so-called interruption at intervals of a fixed time. When not performing the adaptation, the system skips directly to the step 212 for the speech recognition. Operations of up to the step 214 therefrom are similar to those the foregoing example.

Fig. 27 shows another embodiment of the speech recognition system which performs the speaker adaptive processing without teacher. The illustrated system is intended to register users having high use frequencies and for the speaker speech to switch to a pointer table oriented thereto. In the case of a non-

registered speaker, the system switches to a general pointer table.

As in the above case, when the system starts its operation in the step 201, the system first boots in the step 202. When the system boots, the system inputs speech data in the step 1701. In a step 1801, the system performs feature analysis for speaker identification (such as high frequency component analysis). Thereby feature data 1851 for the speaker identification is acquired.

In a step 1802, the system perform the speaker identification with use of the feature data 1851 for the speaker identification and identification information 1852. For example, the speaker feature is previously registered as the identification information 1852 so that the system can judges the speaker by identifying the presence of a registered pattern closest to the speaker feature data 1851. A processing channel is provided for each of speakers judgeable in the speaker identifying operation (step 1802). Processing (program) is the same for the respective processing channels, but parameters such as access pointer tables unique to the above speakers and general speakers are provided therefor. Since the judgement of enabled or disabled adaptation varies from speaker to speaker (from parameter to parameter), the adaptive processing is represented as separated by the respective speakers in Fig. 27.

In this example, parameter sets corresponding to the number of registered speakers and a default (standard pattern for general speaker) are used. If the number of registered speakers is two, then three channel parameter sets becomes necessary. Each parameter set includes at least a pointer table.

In the step 212 and subsequent steps, the system performs recognizing operation similar to in the above example, except that the pointer table 420 of the used global intermediate table 400 is provided for each of the speakers. The global intermediate table 400 is used commonly to all the speakers. This is because the memory capacity necessary for formation of various sorts of tables can be suppressed. In this connection, the global intermediate table can be provided as separated for the respective speakers, in which case, however, the area of the memory occupied by the global intermediate tables become enormous.

Fig. 28 shows yet another embodiment of the speech recognition system which executes the speaker adaptive processing without teacher. As in Fig. 27, in the present system, users who use the system especially frequently are registered, and change-over is carried out to the parameter set oriented to the speaker with respect to the speech of the speaker. In this embodiment, in particular, the number of registered speakers is limited to a fixed value to consider a use frequency.

As in the foregoing embodiment, when the system starts its operation in the step 201, the system first boots in the step 202. After the system boots, the system input speech data in the step 1701. In the step 1801, the system performs feature analysis (such as high frequency component analysis) for speaker identification. The system speaker identification with use of analyzed feature data 1851 for speaker identification in the step 1802. For it the identification information 1852 is used. This can be realized, for example, by previously registering the speaker feature and selecting a registered pattern closest thereto. In the speaker identification 1802, the processing channels are selected. In the respective processing channels, the processing program is the same but the pointer tables to be used therein are different. Since the judgement of enabled or disabled adaptation varies from speaker feature to speaker feature, the adaptive processing is represented as separated for the respective speakers in Fig. 28. The above respect is exactly the same as in Fi. 27.

In the example of Fig. 28, in particular, the system modifies identification information in a step 1901. In this case, in addition to the information used in Fig. 27, a table (speaker management table) listing the use frequencies of the registered speakers as management information is used to limit the number of registered users to a fixed value. After performing

the above operation, the system performs exactly the same operations as those of the procedure explained in Fig. 27.

Details of the above identification information modifying operation (step 1901) will be explained in connection with Figs. 29 and 30. Shown in Fig. 29 is the structure the management table (which will also be referred to simply the speaker management table) 500 relating to speaker management in the identification information 1852. In this case, the table structure has a registered speaker column 501, a use frequency column 502 and a column 503 of pointer (data pointer) to the pointer table 420. The data of these columns can sorted in an descending order of the user frequencies of the registered speakers. Such a management table 500 is unnecessary for a one-channel data set but becomes necessary for plural-channel data sets. When the structure is fixed as in Figs. 25 and 27 (when sorting is unnecessary), however, it is not necessarily required to convert data to such a table and is only required to have information such as the data pointer merely as reference data.

In the step 1901 of identification information modification in Fig. 28, for example, the modification and change of the table structure must be made depending on the frequency information. Which will be briefly explained. This processing procedure is shown in Fig. 30. When the system starts its

operation in a step 2001, the system first judges the presence or absence of the speaker corresponding to the identified speaker in the list (speaker management table 500) in a step 2002. In the absence of the corresponding speaker in the list, the system replaces the this-time speaker by the registered speaker located on the lowermost line in a step 2003. In the list replacement of the step 2003, the system erases data on the lowermost line, writes ID (which is registration ID in the speech recognition) of the newly-registered speaker in the 'registered speaker' column, and sets the frequency information to a value (e.g., 5) larger than 1. The data pointer inherits the value allocated to the former, or the corresponding pointer table 420 of the global intermediate table 400 is set (initialized) to a table corresponding to a standard pattern.

In a step 2004, the system updates the frequency information. That is, when the speaker selected by the speaker identification is one of the registered speakers, the system increments the frequency information of the registered speaker, and the system decrements the frequency information of the registered speakers other than the corresponding registered speaker. As a result, the frequency information of the speaker not using the system so frequently after the initialization becomes smaller than that the initialized frequency value (5 in this

example) and the less-frequent speaker is located lower than the just-initialized speaker. That is, it can be avoided that the speaker initialized and just registered be immediately erased from the list.

In a step 2005, the system performs sorting operation of use frequency over the list in response to a change in the order involved by the above operation. There are so many ways of sorting. For example, since a sequential relation in the decremented group is kept, such a bubble sort as will be explained later in connection with Fig. 33 may be efficiently executed. That is, it is only required to subject the initialized list and incremented list alone to the bubble sorting, which manner is shown in Figs. 31 to 33.

Fig. 31 shows an exemplary list newly replaced through the initialization for explaining its replacing operation. In this case, the bubble sorting is carried out sequentially from the lowermost line. Fig. 32 shows an already-existing list. In this case, the bubble sorting is carried out sequentially from a list presence position. Frequency information of lists other than the target list are decremented by one so that the order of the target list is moved necessarily in the upper direction. Accordingly it is not required to operate the lists other than the target list.

The above procedure is shown in Fig. 33 in the form of a flowchart, in which case the sorting operation is shown. When the system starts its

operation in a step 2101, the system selects a list to be sorted in a step 2102. The list corresponds to the list of the speaker in question. In a step 2103, the system compares the frequency information of the target list with that of the just-upper list. When the sequential relation is correct, the system terminates its operation in a step 2105. When the sequential relation is not correct, the system replaces the target list with the just-upper list and returns to the step 2103. The above operations are repeated until the sequential relation becomes normal (until the frequency information of the target list becomes smaller than the frequency information of the just-upper list or the target list reaches the uppermost position), at which stage the system terminates the operation in a step 2105.

In accordance with the foregoing embodiment, operations and effects which follow can be obtained.

In the above computation of the output probability, the feature components are linearly quantized with the same scale upon the computation of all the mixture multi-dimensional Gaussian distributions. Therefore, it is required only once per one frame to linearly scalar-quantize the feature vector (integer value corresponding to a floating or fixed point). Further, a difference (feature offset or table offset) between data to be referred to and the first address of the intermediate table belonging to

the data is also common to the respective feature components. Accordingly, the computation of the single Gaussian distribution can be executed by loading the first address of the intermediate table, adding together the first address of the intermediate table and the feature offset, accessing the intermediate table, and accessing the numeric value table. As a result, the computational speed of the output probability can be increased.

In the adaptation, it is unnecessary to rewrite the numeric value table itself. When the pointer table is used, it is unnecessary to rewrite the intermediate table. And it is only required to modify the value of the access pointer in the pointer table in response to a change of the dispersion or average caused by the adaptation. As a result, the speed of the adaptive processing can also be made higher.

The numeric value table is generally stored in an external memory. However, the system makes access to the numeric value table not immediately after acquiring one data address in the numeric value table through the table access but after previously finding all the data addresses for each multi-dimensional Gaussian distribution. Thus before the system starts the access to the numeric value table, the system can prefetch the data addresses into the data cache 117. Accordingly, upon the access to the numeric value table, the system can get a cache hit and a cache miss

in the access to the numeric value table can be avoided.

From the foregoing, when an output probability is computed for speech recognition, upon a series of memory accesses for data reference, the system can obtain the numeric value of the Gaussian distribution without generation of a cache miss by three data loads and one addition (for address computation). Even when the frequency of the access operation to the intermediate table is increased, the computation speed of the output probability can be remarkably increased.

Further, the global intermediate table 400 is employed which allows extraction of the intermediate table 401, 402 uniquely associated with the dispersion and average of the one-dimensional Gaussian distribution, the first address of the intermediate table 401, 402 extracted from the global intermediate table 400 is specified by the access pointer in the pointer table 420, and the access location to the extracted intermediate table is specified by the feature offset obtained through the linear quantization of the feature component. Therefore, even when the dispersion or average is changed by the adaptation, the rewriting of the intermediate table is not required, the system can cope with the change merely by rewriting the value of the access pointer associated with the change in the pointer table, thus realizing higher

layers of adaptive processing.

Further, the value of the access pointer has a correlation with the dispersion and average. Thus when the dispersion and average are changed by the adaptation, the operation of changing the value of the access pointer is correspondingly simplified.

The speed of the speaker adaptive processing can be made higher by previously setting a plurality of sets of access pointer tables and switchingly using the access pointer tables depending on the speaker adaptive processing.

The invention made by the present inventor has been explained in detail in connection with the embodiments thereof. However, the present invention is not limited to the specific examples but may be modified in various ways in a range not departing from its gist.

For example, the data processing system is not limited to the portable information terminal device. The portable telephone function may be omitted. The system may also be implemented under control of a personal computer system.

The arrangement of the data processor is not limited to the arrangement of Fig. 2. The 'data processor' generally refers to a processor known as a microprocessor or a microcomputer. The data processor is a circuit which fetches an instruction and decodes the fetched instruction to perform computational

control operation. And the data processor is only required to have a CPU (central processing unit). It is further preferable that the data processor incorporate a data cache memory or a high speed RAM. The global intermediate table, pointer table or the like is resident in the high speed RAM incorporated in the data processor.

The computer-readable medium having the program for the output probability computation for the HMM speech recognition stored therein may be a magnetic storage medium such as a floppy disk, magnetic tape or hard disk, an optical storage medium such as a CD-ROM or MO, a semiconductor recording medium such as a memory card, or any medium other than the above.

INDUSTRIAL APPLICABILITY

The present invention can be widely applied to a speech recognition techniques using HMM. For example, the present invention is directed to a technique for effectively being able to be applied to implementation of the speech recognition of a portable information terminal device which is controlled under control of a microcomputer and driven by a battery. Further, the program for output probability computation for the speech recognition of the present invention may be used by loading the program into a computer such as a personal computer through a computer-readable recording medium or a communication line.

CLAIMS

1. A data processing system wherein a data processor refers to an intermediate table and a numeric value table for HMM speech recognition with respect to a feature vector to compute an output probability represented by a mixture multi-dimensional Gaussian distribution, said numeric value table has a region which contains numeric values of a plurality of types of one-dimensional Gaussian distributions, said intermediate table has a region which is selected on the basis of a value of linear quantization for a value of a feature component of said feature vector and which contains address information indicative of a location of the value of said numeric value table, said data processor linearly quantizes the value of said feature component, selects the intermediate table based on an access pointer for each feature component, acquires the address information from said selected intermediate table on the basis of said linearly-quantized value, refers to the numeric value table with use of the acquired address information, and computes said output probability on the basis of the value referred to from the numeric value table.

2. A data processing system as set forth in claim 1, having a region for formation of an access pointer table which contains said access pointers arranged for the feature components for the respective multi-dimensional Gaussian distributions of a mixture

multi-dimensional Gaussian distribution, and wherein said data processor selects the intermediate tables with use of the access pointer of said access pointer table.

3. A data processing system as set forth in claim 1 or 2, wherein said entire distribution based on each of said one-dimensional Gaussian distributions is represented by 2^N numeric values, and the quantized value of said feature component correspond to upper N bits of said values.

4. A data processing system as set forth in claim 1 or 2, wherein said data processor repetitively refers to said numeric value table for each feature component to compute the values of the multi-dimensional Gaussian distributions, and repetitively computes the values of the multi-dimensional Gaussian distributions by a predetermined number of times to compute the output probability represented by the mixture multi-dimensional Gaussian distribution.

5. A data processing system as set forth in claim 4, wherein said intermediate table has a region which contains said address information in a range of a multiple of a dispersion with an average position of the one-dimensional Gaussian distribution as a start point and a reference of said numeric value table and also has a region outside of said region which contains distance information from said average, said data processor repetitively refers to said numeric value

table for each feature component to compute the value of the multi-dimensional Gaussian distribution in such a manner, when information referred to from said numeric value table is said distance information, the data processor accumulates it and, when the accumulated value exceeds a predetermined value, the data processor stops the computation of the corresponding multi-dimensional Gaussian distribution.

6. A data processing system as set forth in claim 5, wherein said intermediate table has a region which contains a fixed value outside of said distance information, and said data processor, when referring to said fixed value from said intermediate table, stops the computation of the corresponding multi-dimensional Gaussian distribution being currently processed.

7. A data processing system wherein a data processor refers to a global table and a numeric value table for HMM speech recognition with respect to a feature vector to compute an output probability represented by a mixture multi-dimensional Gaussian distribution, said numeric value table has a region which contains numeric values of a plurality of types of one-dimensional Gaussian distributions having a mutually identical average and different dispersions, said global table has a region which contain a plurality of sets of X-direction arrays in a Y direction for each distribution of said numeric value table, said X-direction arrays have a region which

contains address information indicative of a location of the value of said numeric value table at a location selected on the basis of a value of linear quantization for a value of a feature component of said feature vector, said data processor linearly quantizes the value of said feature component, extracts the intermediate table from said global table according to the value of an access pointer for each feature component taking into consideration a dispersion upon selection of the plurality of sets of X-direction arrays in the Y direction and taking into consideration an average upon determination of a first location of the X-direction array, acquires the address information on the basis of said linearly-quantized value with the first location of said extracted intermediate table as a start point, refers to the numeric value table with use of the acquired address information, and computes said output probability on the basis of the value referred to from the numeric value table.

8. A data processing system as set forth in claim 7, having a region for formation of an access pointer table which contains said access pointers arranged for the feature components for the respective multi-dimensional Gaussian distributions of a mixture multi-dimensional Gaussian distribution, and wherein said data processor selects the intermediate tables with use of the access pointer of said access pointer table.

9. A data processing system as set forth in claim 8, wherein said data processor, when both or either one of the average and dispersion of the mixture multi-dimensional Gaussian distribution is changed by adaptation, correspondingly changes the value of the access pointer of said access pointer table.

10. A data processing system as set forth in claim 8, having a region for formation of a plurality of sets of said access pointer tables, and wherein said data processor identifies a speaker and uses the access pointer table corresponding to its identified result.

11. A data processing system as set forth in claim 10, wherein said speaker identification is carried out on the basis of a state of a switch for clarification of the speaker.

12. A data processing system as set forth in claim 10, having a region for formation of a management table showing a relation between said access pointer table and speaker, and wherein said data processor performs said speaker identification on the basis of a comparison result between identification feature information indicative of a feature of the speaker and previously registered and an actual speech feature analysis result, when determining that the identified speaker is one of speakers registered in said management table, the data processor refers to the access pointer table of the corresponding registered speaker.

13. A data processing system as set forth in claim 12, wherein said data processor limits the number of speakers registerable in said management table to a fixed value, adds information about a user frequency for each registered speaker to said management table, increments, when determining one of the registered speakers as a result of the speech feature analysis, the use frequency of the registered speaker corresponding to the analysis result, decrements the use frequencies of the registered speakers other than the speaker corresponding to the analysis result, deletes, when the speech feature analysis result indicates the speaker is not registered, the registered speaker having a lowest use frequency from said management table, and instead adds the not-registered speaker to the management table.

14. A data processing system as set forth in claim 8, having a plurality of speech input channels and having a region for formation of said access pointer table for each speech input channel, and wherein said data processor uses the access pointer table independently with respect to said plurality of speech input channels to enable parallel speech recognition.

15. A data processing system as set forth in claim 7 or 8, wherein said data processor linearly quantizes all feature components of a feature vector, computes a feature offset from a first location of the

extracted intermediate table on the basis of a product of said quantized value and an address amount of a single array element of said X-direction array, and thereafter refers to the intermediate table on the basis of said access pointer and feature offset for each multi-dimension mixture Gaussian distribution to refer to the numeric value table.

16. A data processing system as set forth in claim 15, wherein said entire distribution based on each of said one-dimensional Gaussian distributions is represented by 2^N numeric values, and the quantized value of said feature component correspond to upper N bits of said values.

17. A data processing system as set forth in claim 16, wherein said data processor repetitively refers to said numeric value table for each feature component to compute the values of the multi-dimensional Gaussian distributions, and repetitively computes the values of the multi-dimensional Gaussian distributions by a predetermined number of times to compute the output probability represented by the mixture multi-dimensional Gaussian distribution.

18. A data processing system as set forth in claim 17, wherein each of said X-direction arrays has a region which contains address information in a range of a multiple of a dispersion with an average position of the one-dimensional Gaussian distribution as a start point and a reference of said numeric value table and

also has a region outside of said region which contains distance information from said average, said data processor repetitively refers to said numeric value table for each feature component to compute the value of the multi-dimensional Gaussian distribution in such a manner, when information referred to from said numeric value table is said distance information, the data processor accumulates it and, when the accumulated value exceeds a predetermined value, the data processor stops the computation of the corresponding multi-dimensional Gaussian distribution.

19. A data processing system as set forth in claim 18, wherein each of said Y-direction arrays has a region which contains a fixed value outside of said distance information, and said data processor, when referring to said fixed value from said intermediate table, stops the computation of the corresponding multi-dimensional Gaussian distribution being currently processed.

20. A method for computing an output probability of a mixture Gaussian HMM, comprising the steps of:

using a numeric value table which contains numeric values of distributions based on a plurality of types of one-dimensional Gaussian distributions for HMM speech recognition with respect to a feature vector;

using an intermediate table which contains address information indicative of a location of a value of said numeric value table corresponding to a

linearly-quantized value of a value of a feature component of said feature vector in a region selected based on the quantized value; and

linearly quantizing the value of said feature component, selecting the intermediate table on the basis of an access pointer of each feature component, acquiring address information from said intermediate table selected on the basis of said linearly-quantized value, referring to the numeric value table with use of the acquired address information, and computing the output probability represented by a mixture multi-dimensional Gaussian distribution.

21. A method for computing an output probability of a mixture Gaussian HMM as set forth in claim 20, wherein the selection of said intermediate table is carried out with use of an access pointer table which contains said access pointers arranged therein for the respective feature components of the respective multi-dimensional Gaussian distributions for a mixture multi-dimensional Gaussian distribution.

22. A method for computing an output probability of a mixture Gaussian HMM, comprising the steps of:

using a numeric value table which contains numeric values of distributions based on a plurality of types of one-dimensional Gaussian distributions having an identical average and mutually different dispersions for HMM speech recognition with respect to a feature vector;

using a global table which contains a plurality of sets of X-direction arrays in a Y direction for each distribution in said numeric value table, said X-direction arrays containing address information indicative of a location of a value of said numeric value table corresponding to a linearly-quantized value of a value of a feature component of said feature vector in a region selected based on the quantized value; and

linearly quantizing the value of said feature component, extracting the intermediate table from said global table on the basis of the value of an access pointer of each feature component taking into consideration a dispersion upon selection of the plurality of sets of X-direction arrays in the Y direction and taking consideration an average upon determination of a first location of the X-direction array, acquiring said address information on the basis of said linearly-quantized value with the first location of said extracted intermediate table as a start point, referring to the numeric value table with use of the acquired address information, and computing the output probability represented by a mixture multi-dimensional Gaussian distribution.

23. A method for computing an output probability of a mixture Gaussian HMM as set forth in claim 22, wherein the extraction of said intermediate table is carried out with use of an access pointer table which

contains said access pointers arranged therein for the respective feature components of the respective multi-dimensional Gaussian distributions for a mixture multi-dimensional Gaussian distribution.

24. A method for computing an output probability of a mixture Gaussian HMM as set forth in claim 23, wherein, when both or either one of the average and dispersion of the mixture multi-dimensional Gaussian distribution is changed by adaptation, the value of the access pointer of said access pointer table is correspondingly changed.

25. A recording medium readable by a computer and having a program recorded therein, wherein said program is executed under control of the computer, said program uses a numeric value table which contains numeric values of distributions based on a plurality of types of one-dimensional Gaussian distributions to input a feature vector for HMM speech recognition;

uses an intermediate table which contains address information indicative of a location of a value of said numeric value table corresponding to a linearly-quantized value of a value of a feature component of said feature vector in a region selected based on the quantized value;

uses an access pointer table which contains access pointers arranged therein for the respective feature components of the multi-dimensional Gaussian distributions of a mixture multi-dimensional Gaussian

distribution; and

linearly quantizes the value of said feature component, selects the intermediate table on the basis of the access pointer of each feature component in said access pointer table, acquires address information from said intermediate table selected on the basis of said linearly-quantized value, refers to the numeric value table with use of the acquired address information, and computes the output probability represented by the mixture multi-dimensional Gaussian distribution.

26. A recording medium readable by a computer and having a program recorded therein, wherein said program is executed under control of the computer, said program uses a numeric value table which contains numeric values of distributions based on a plurality of types of one-dimensional Gaussian distributions having an identical average and mutually different dispersions to input a feature vector for HMM speech recognition;

uses a global table which contains a plurality of sets of X-direction arrays in a Y direction for distributions in said numeric value table, each of said X-direction arrays containing address information indicative of a location of a value of said numeric value table corresponding to a linearly-quantized value of a value of a feature component of said feature vector at a position selected based on the quantized value;

uses an access pointer table which contains

access pointers arranged for the respective multi-dimensional Gaussian distributions of the mixture multi-dimensional Gaussian distribution taking into consideration a dispersion upon selection of the plurality of sets of X-direction arrays in the Y direction and taking into consideration an average upon determination of a first location of the X-direction array; and

linearly quantizes the value of said feature component, extracts the intermediate table from said global table on the basis of the value of the access pointer in said access pointer table, acquires address information on the basis of said linearly-quantized value with the first location of said extracted intermediate table as a start point, refers to the numeric value table with use of the acquired address information, and computes the output probability represented by the mixture multi-dimensional Gaussian distribution.

27. A recording medium readable by a computer as set forth in claim 23, wherein, when both or either one of the average and dispersion of the mixture multi-dimensional Gaussian distribution is changed by adaptation, said program correspondingly changes the value of the access pointer of said access pointer table.

28. A data processing system as set forth in claim 1 or 7, having a battery for supplying an

operational power, and wherein said data processor operates on said battery as its operating power source and has a power consumption of 1W or less.

E 6055

09/890703

FIG. 1

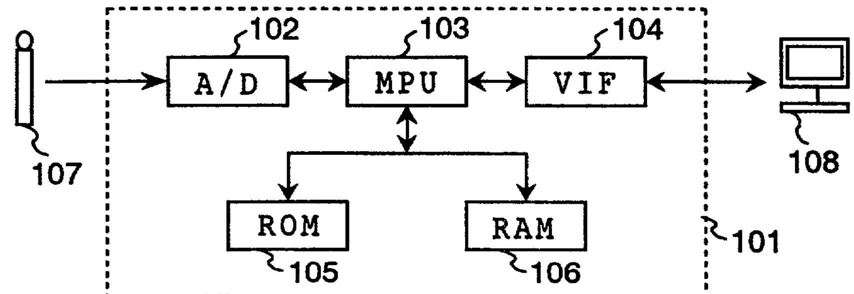


FIG. 2

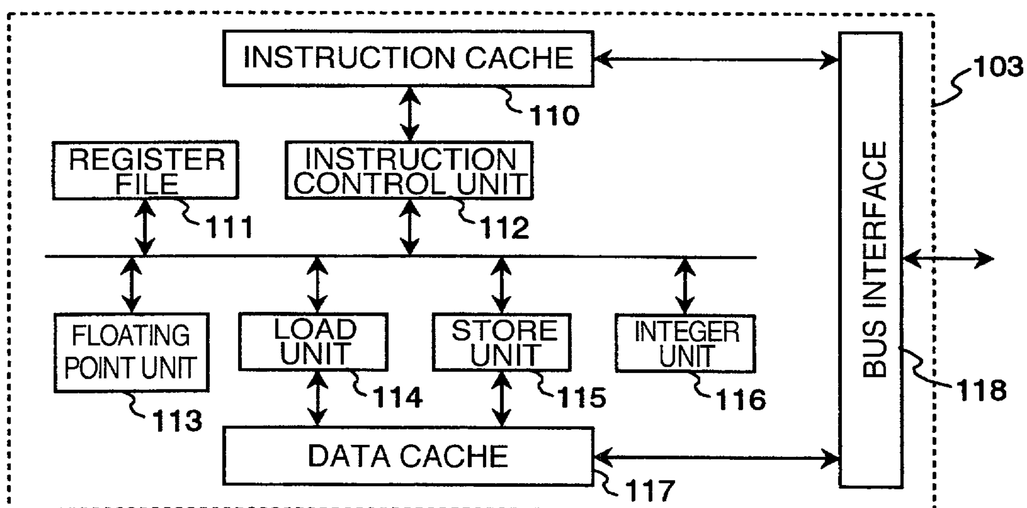


FIG. 3

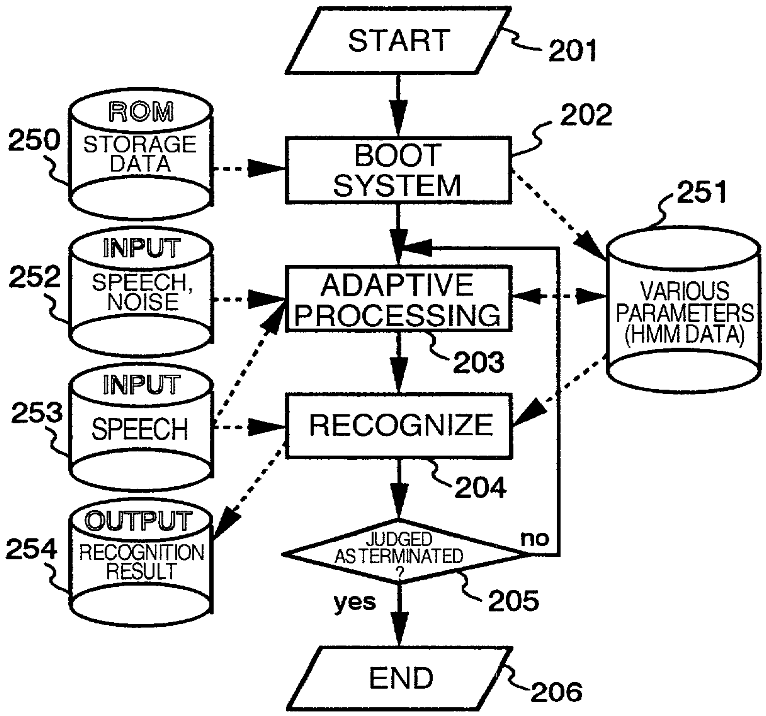

START
201
ROM
STORAGE DATA
250
BOOT SYSTEM
202
251
INPUT
SPEECH, NOISE
252
ADAPTIVE PROCESSING
VARIOUS PARAMETERS (HMM DATA)
203
INPUT
SPEECH
253
RECOGNIZE
204
OUTPUT
RECOGNITION RESULT
254
JUDGED AS TERMINATED ?
no
yes
205
END
206

FIG. 4

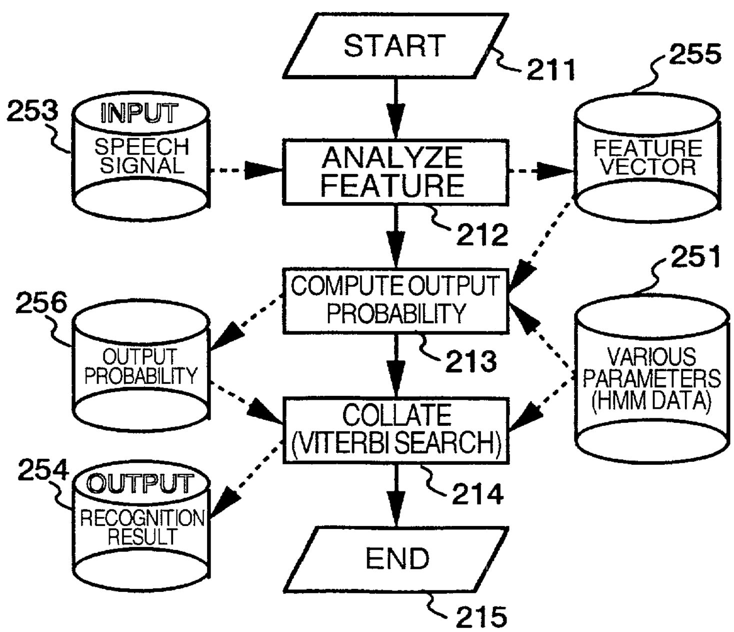

START
211
255
INPUT
SPEECH SIGNAL
253
ANALYZE FEATURE
FEATURE VECTOR
212
251
COMPUTE OUTPUT PROBABILITY
256
OUTPUT PROBABILITY
213
VARIOUS PARAMETERS (HMM DATA)
COLLATE (VITERBI SEARCH)
254
OUTPUT
RECOGNITION RESULT
214
END
215

FIG. 5

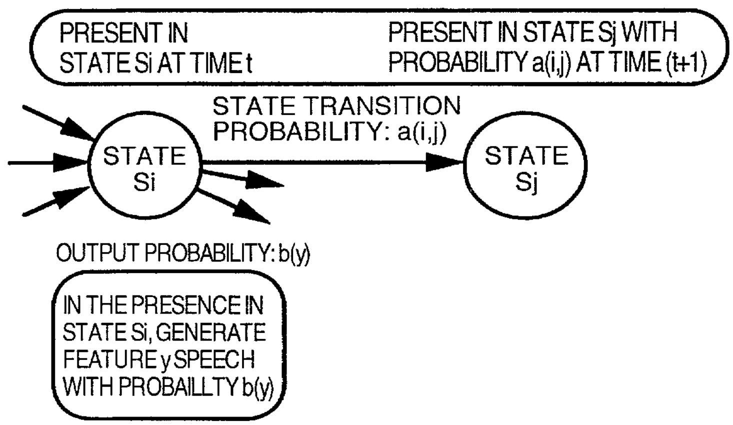

PRESENT IN
STATE Si AT TIME t
PRESENT IN STATE Sj WITH
PROBABILITY a(i,j) AT TIME (t+1)
STATE TRANSITION
PROBABILITY: a(i,j)
STATE
Si
STATE
Sj
OUTPUT PROBABILITY: b(y)
IN THE PRESENCE IN
STATE Si, GENERATE
FEATURE y SPEECH
WITH PROBAILLTY b(y)

FIG. 6

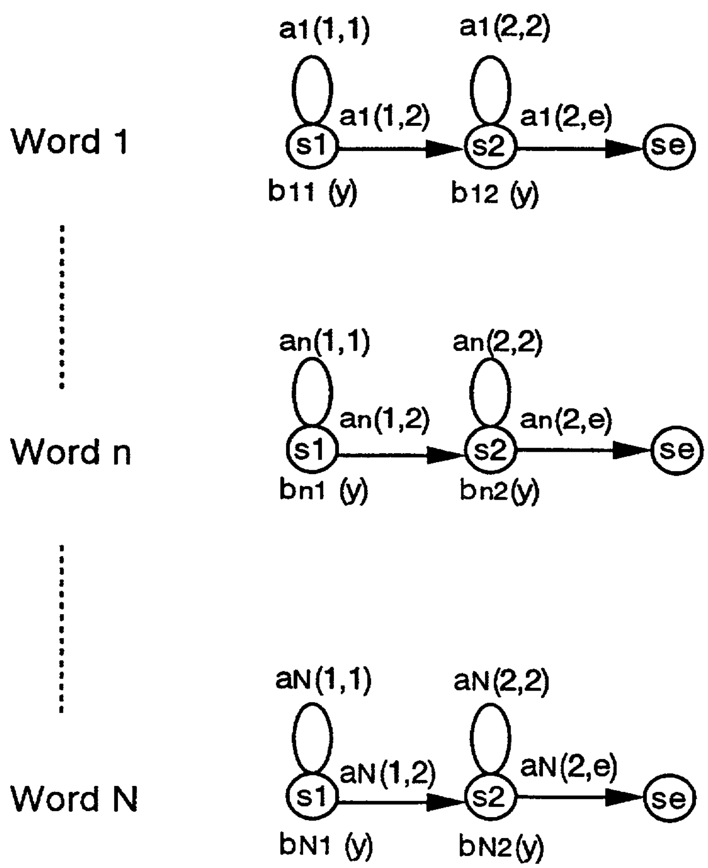

Word 1
a1(1,1)
a1(2,2)
a1(1,2)
a1(2,e)
s1
s2
se
b11 (y)
b12 (y)
Word n
an(1,1)
an(2,2)
an(1,2)
an(2,e)
s1
s2
se
bn1 (y)
bn2(y)
Word N
aN(1,1)
aN(2,2)
aN(1,2)
aN(2,e)
s1
s2
se
bN1 (y)
bN2(y)

FIG. 7

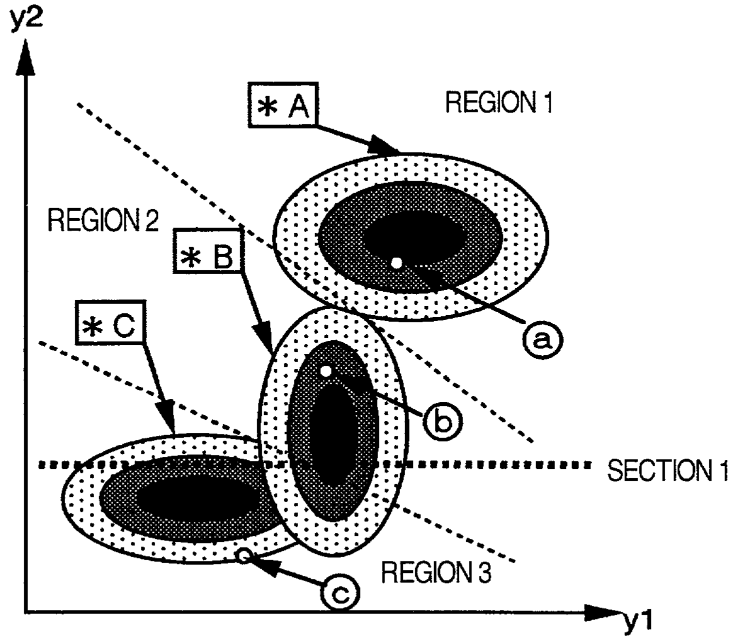

y2
∗ A
REGION 1
REGION 2
∗ B
∗ C
a
b
SECTION 1
REGION 3
c
y1

FIG. 8

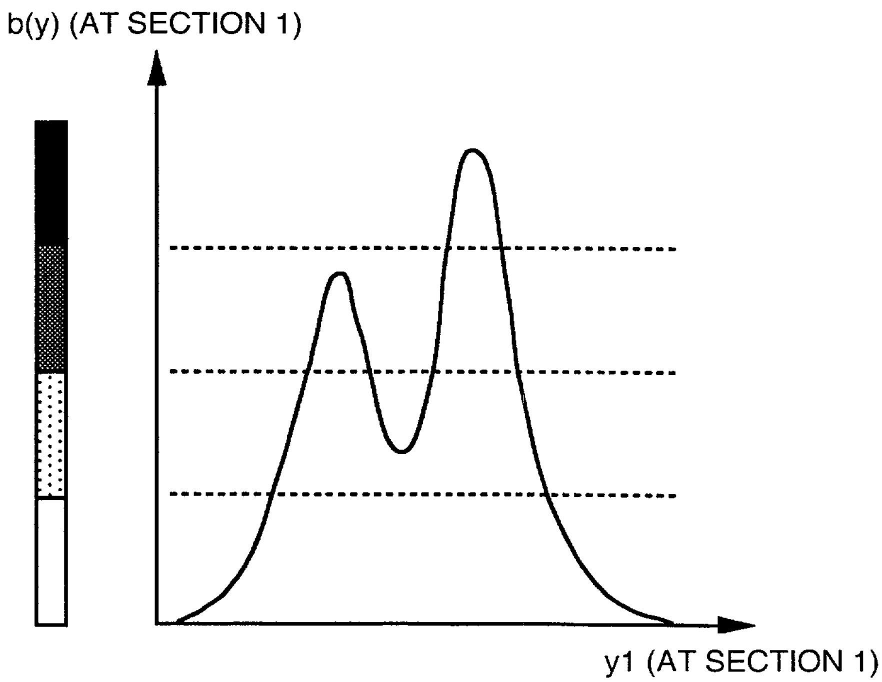

b(y) (AT SECTION 1)
y1 (AT SECTION 1)

FIG. 9

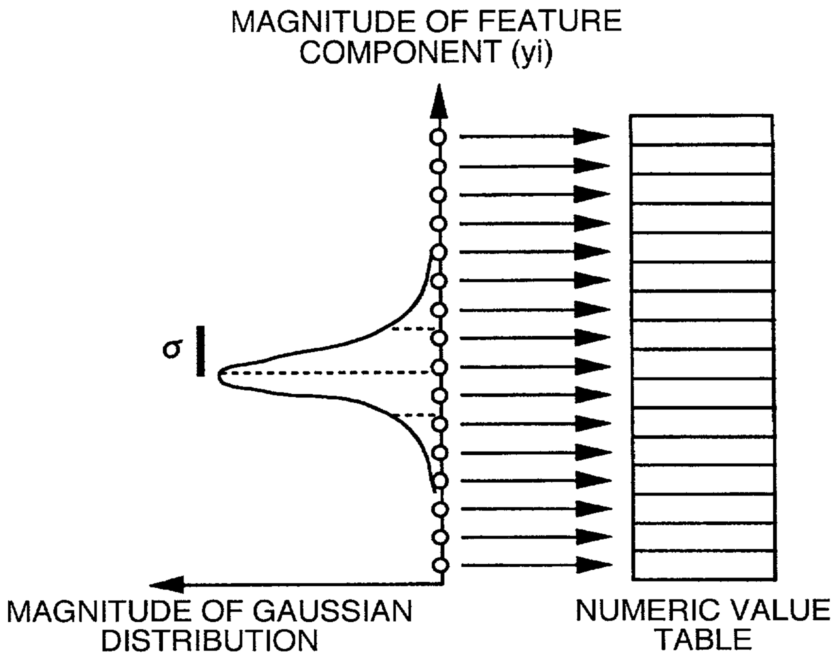

MAGNITUDE OF FEATURE COMPONENT (yi)
σ
MAGNITUDE OF GAUSSIAN DISTRIBUTION
NUMERIC VALUE TABLE

FIG. 10

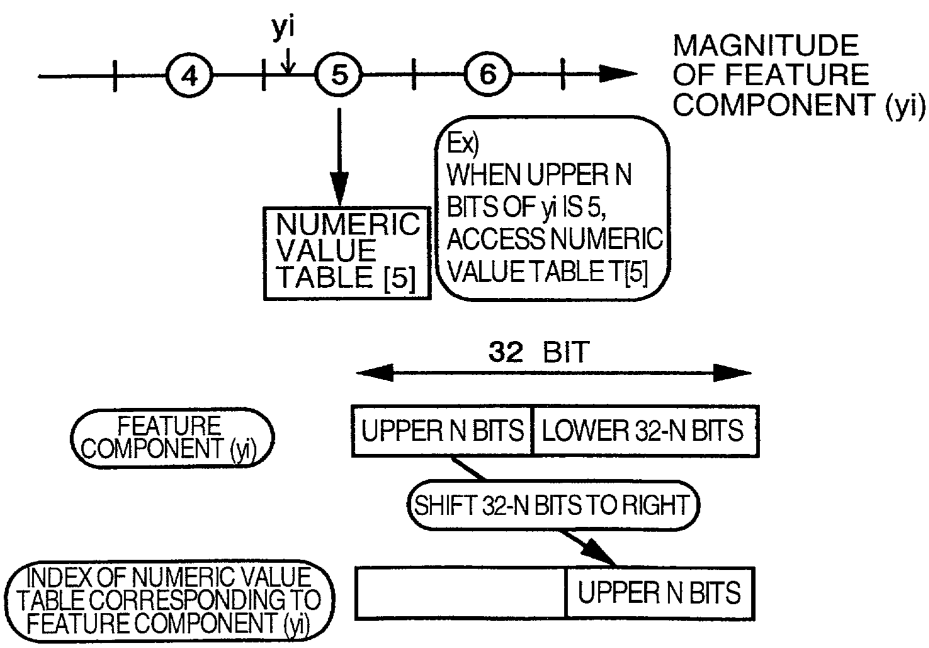
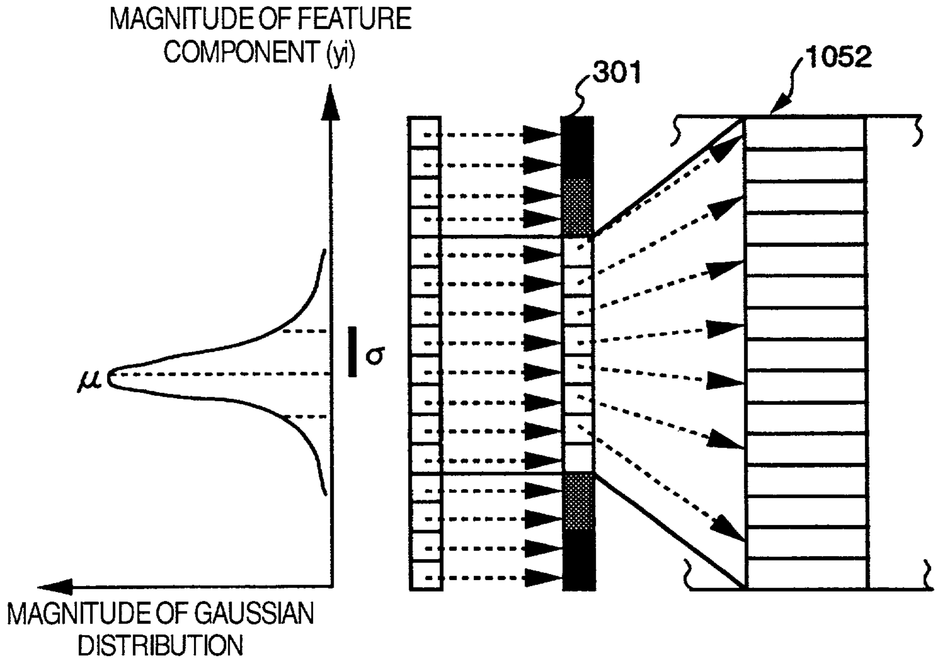
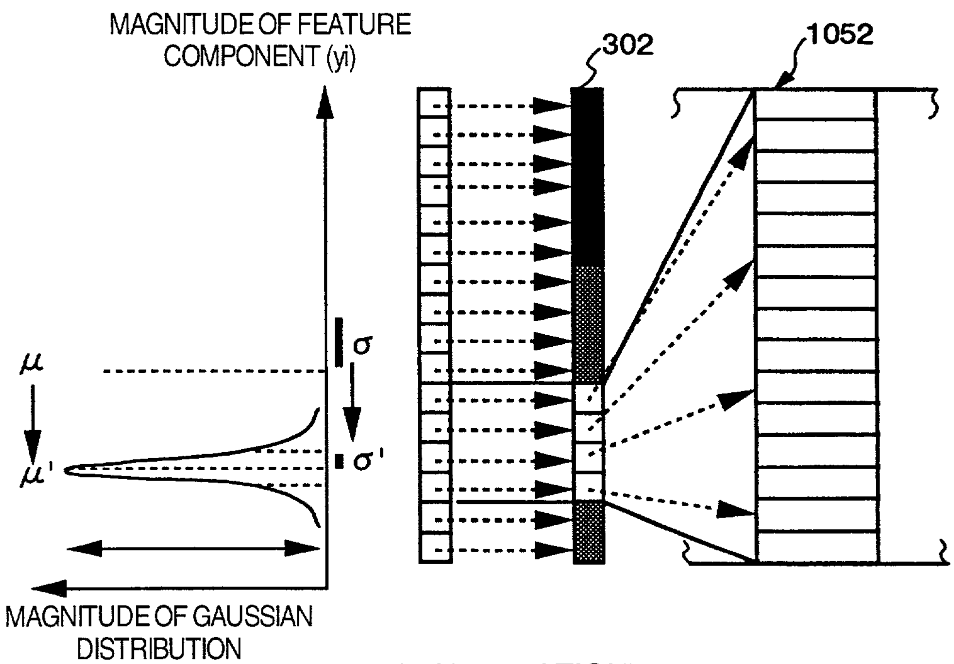

yi
4
5
6
MAGNITUDE OF FEATURE COMPONENT (yi)
NUMERIC VALUE TABLE [5]
Ex) WHEN UPPER N BITS OF yi IS 5, ACCESS NUMERIC VALUE TABLE T[5]
32 BIT
FEATURE COMPONENT (yi)
UPPER N BITS
LOWER 32-N BITS
SHIFT 32-N BITS TO RIGHT
INDEX OF NUMERIC VALUE TABLE CORRESPONDING TO FEATURE COMPONENT (yi)
UPPER N BITS

FIG. 11



[BEFORE ADAPTATION]

FIG. 12



[AFTER ADAPTATION]

FIG. 13

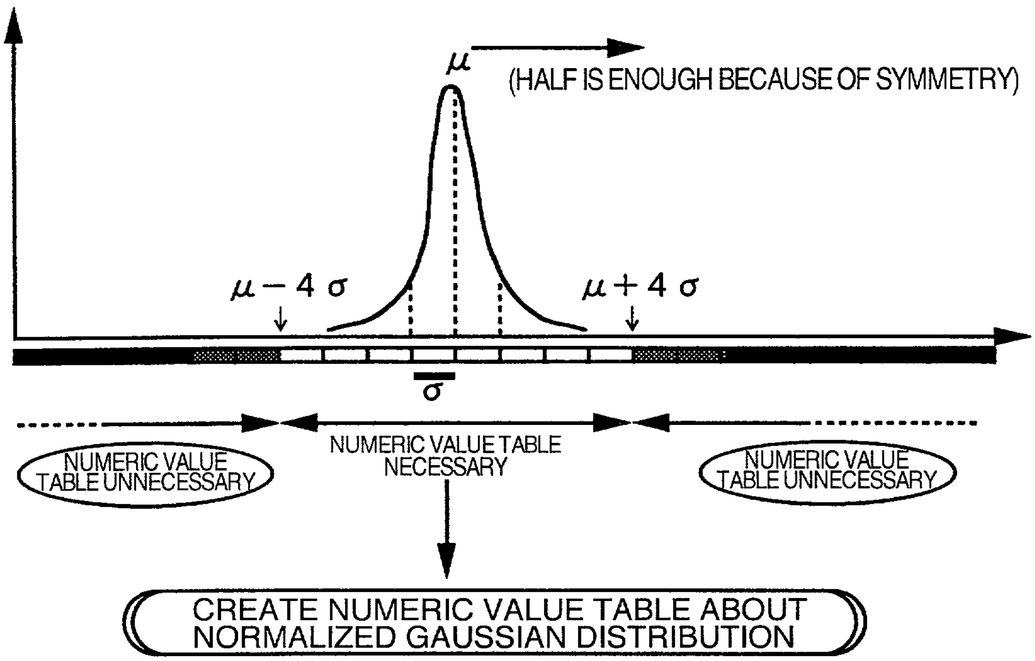

μ
(HALF IS ENOUGH BECAUSE OF SYMMETRY)
$\mu - 4\sigma$
$\mu + 4\sigma$
σ
NUMERIC VALUE TABLE UNNECESSARY
NUMERIC VALUE TABLE NECESSARY
NUMERIC VALUE TABLE UNNECESSARY
CREATE NUMERIC VALUE TABLE ABOUT NORMALIZED GAUSSIAN DISTRIBUTION

FIG. 14

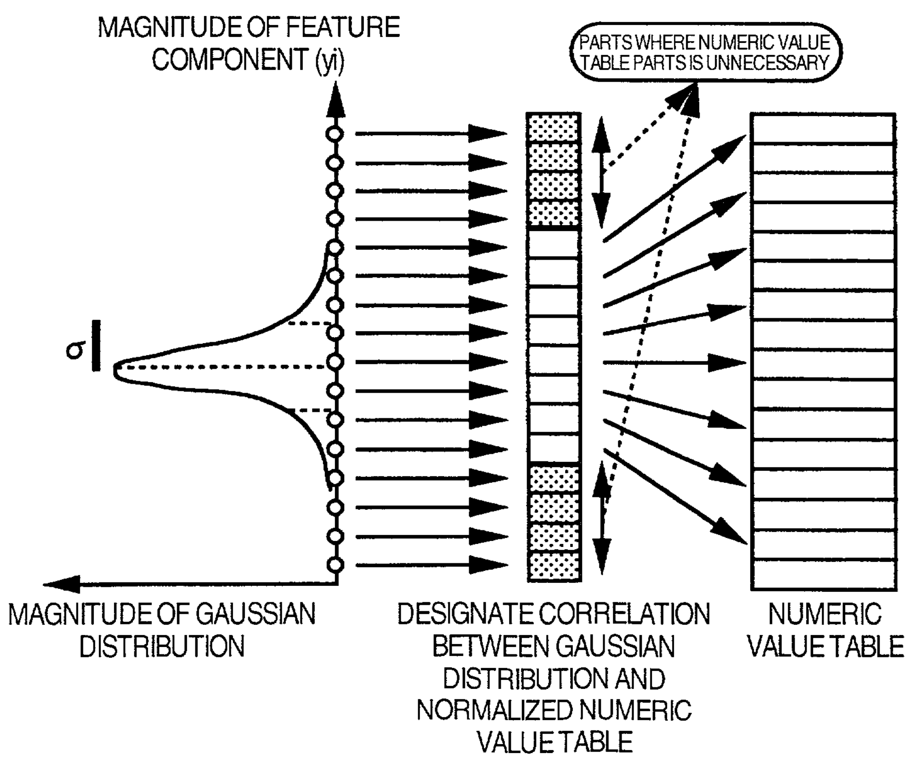

MAGNITUDE OF FEATURE COMPONENT (yi)
PARTS WHERE NUMERIC VALUE TABLE PARTS IS UNNECESSARY
σ
MAGNITUDE OF GAUSSIAN DISTRIBUTION
DESIGNATE CORRELATION BETWEEN GAUSSIAN DISTRIBUTION AND NORMALIZED NUMERIC VALUE TABLE
NUMERIC VALUE TABLE

FIG. 15

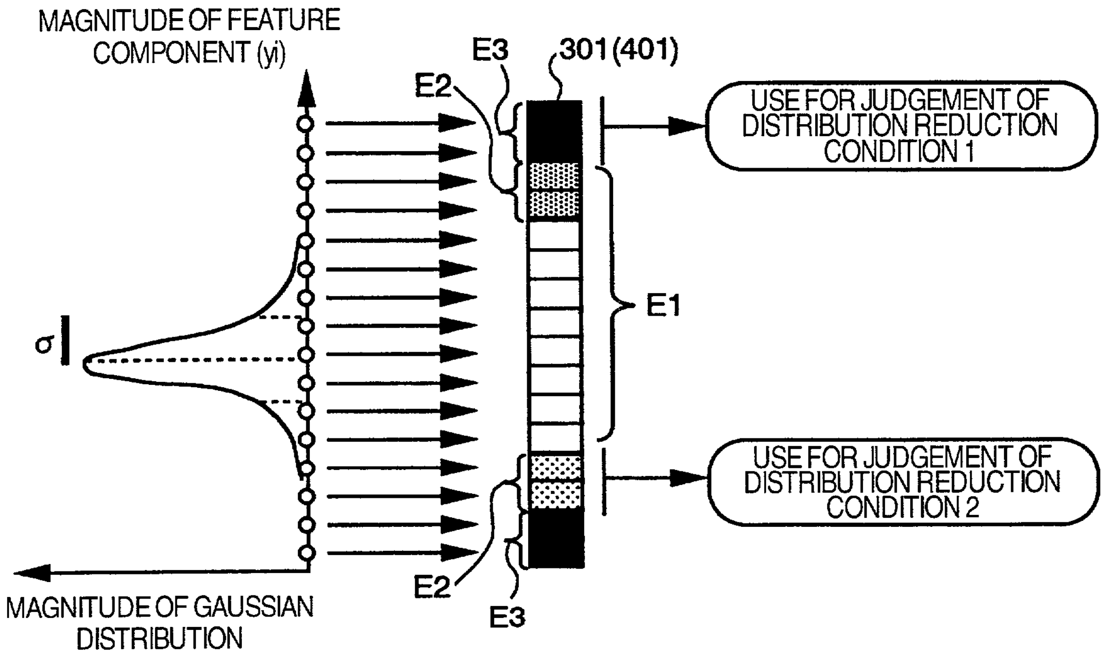
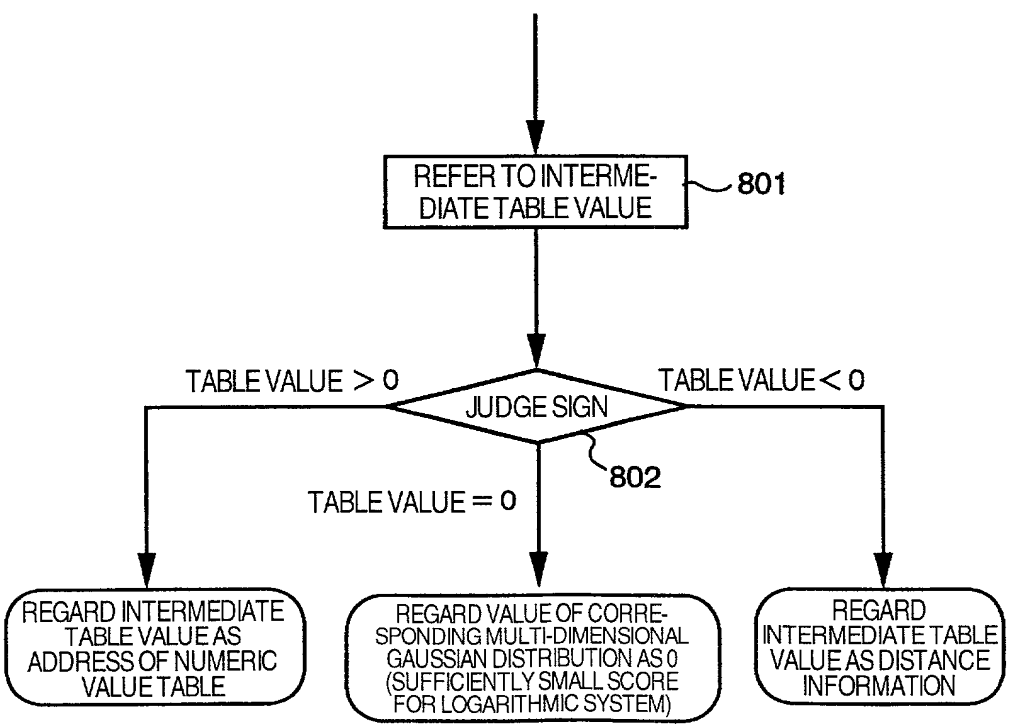

MAGNITUDE OF FEATURE COMPONENT (yi)
E3
301(401)
E2
USE FOR JUDGEMENT OF DISTRIBUTION REDUCTION CONDITION 1
σ
E1
USE FOR JUDGEMENT OF DISTRIBUTION REDUCTION CONDITION 2
E2
E3
MAGNITUDE OF GAUSSIAN DISTRIBUTION

FIG. 16


REFER TO INTERMEDIATE TABLE VALUE
801
TABLE VALUE > 0
JUDGE SIGN
TABLE VALUE < 0
802
TABLE VALUE = 0
REGARD INTERMEDIATE TABLE VALUE AS ADDRESS OF NUMERIC VALUE TABLE
REGARD VALUE OF CORRESPONDING MULTI-DIMENSIONAL GAUSSIAN DISTRIBUTION AS 0 (SUFFICIENTLY SMALL SCORE FOR LOGARITHMIC SYSTEM)
REGARD INTERMEDIATE TABLE VALUE AS DISTANCE INFORMATION

09/890703

FIG. 17

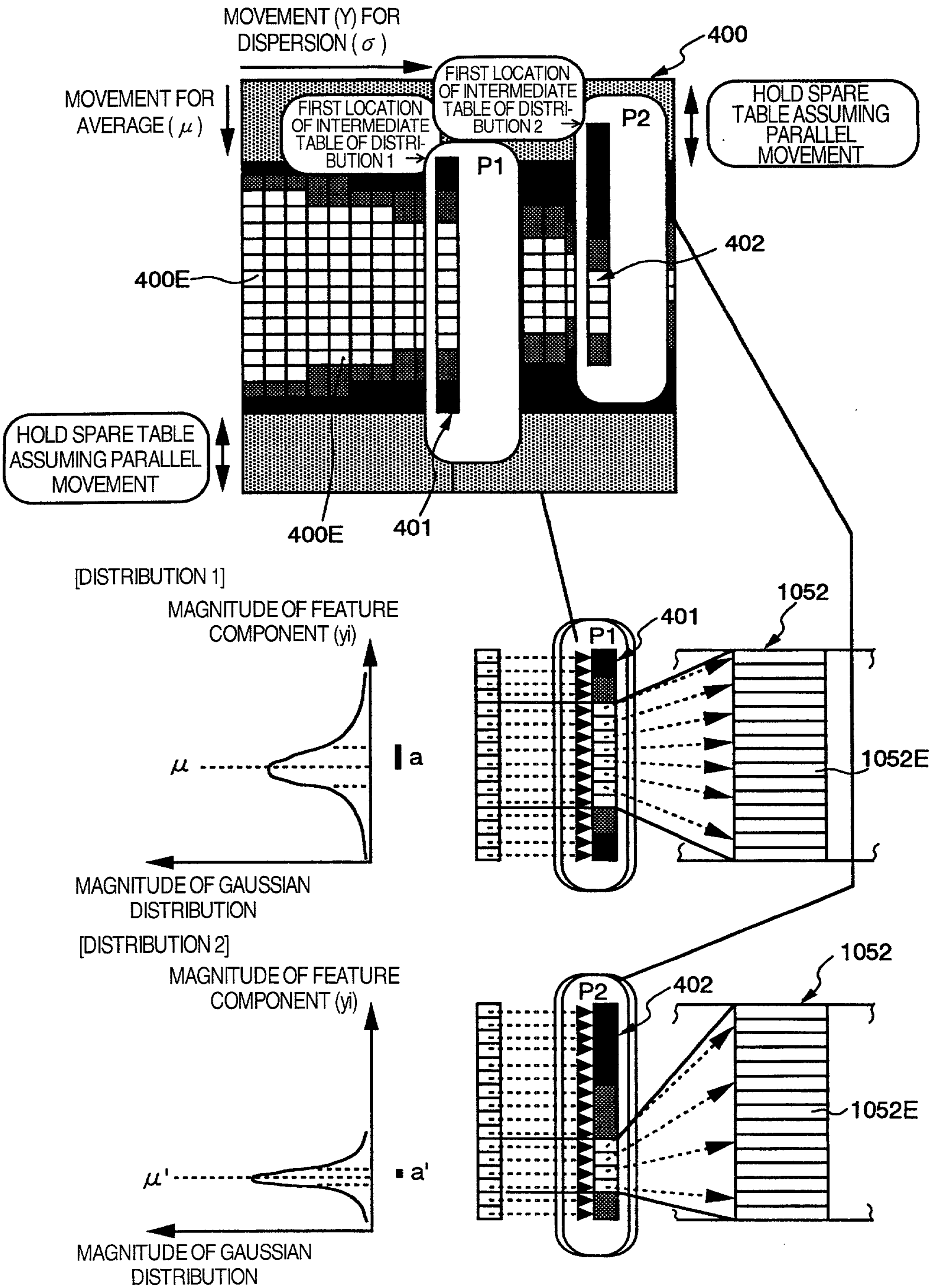
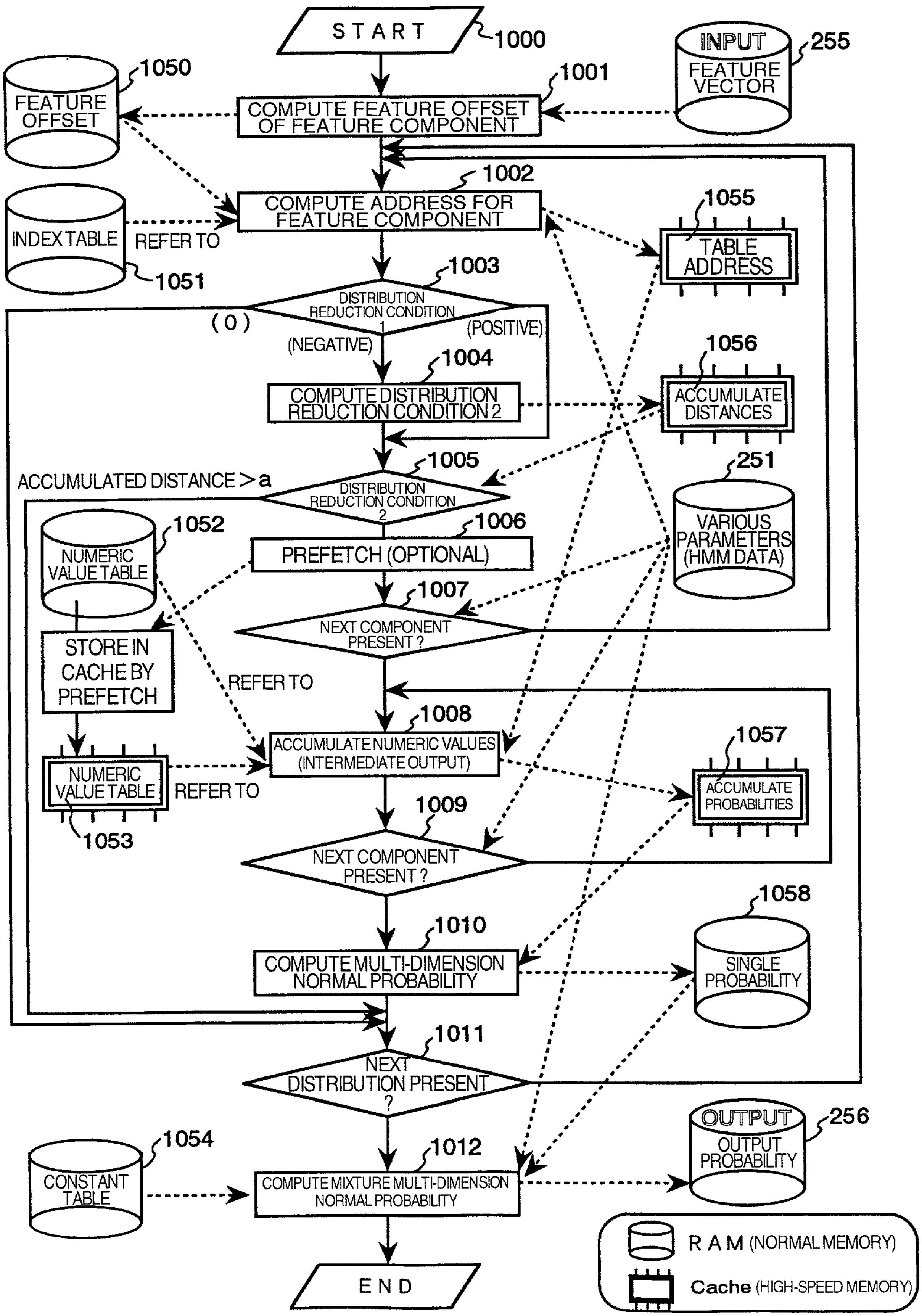


FIG. 18


START
1000
INPUT
FEATURE VECTOR
255
1001
COMPUTE FEATURE OFFSET OF FEATURE COMPONENT
FEATURE OFFSET
1050
1002
COMPUTE ADDRESS FOR FEATURE COMPONENT
INDEX TABLE
REFER TO
1051
1055
TABLE ADDRESS
1003
DISTRIBUTION REDUCTION CONDITION 1
(0)
(NEGATIVE)
(POSITIVE)
1004
COMPUTE DISTRIBUTION REDUCTION CONDITION 2
1056
ACCUMULATE DISTANCES
ACCUMULATED DISTANCE > a
1005
DISTRIBUTION REDUCTION CONDITION 2
251
VARIOUS PARAMETERS (HMM DATA)
1052
NUMERIC VALUE TABLE
1006
PREFETCH (OPTIONAL)
1007
NEXT COMPONENT PRESENT ?
STORE IN CACHE BY PREFETCH
REFER TO
1008
ACCUMULATE NUMERIC VALUES (INTERMEDIATE OUTPUT)
1057
ACCUMULATE PROBABILITIES
NUMERIC VALUE TABLE
REFER TO
1053
1009
NEXT COMPONENT PRESENT ?
1058
1010
COMPUTE MULTI-DIMENSION NORMAL PROBABILITY
SINGLE PROBABILITY
1011
NEXT DISTRIBUTION PRESENT ?
OUTPUT
OUTPUT PROBABILITY
256
1054
CONSTANT TABLE
1012
COMPUTE MIXTURE MULTI-DIMENSION NORMAL PROBABILITY
END
R A M (NORMAL MEMORY)
Cache (HIGH-SPEED MEMORY)

09/890703

FIG. 19

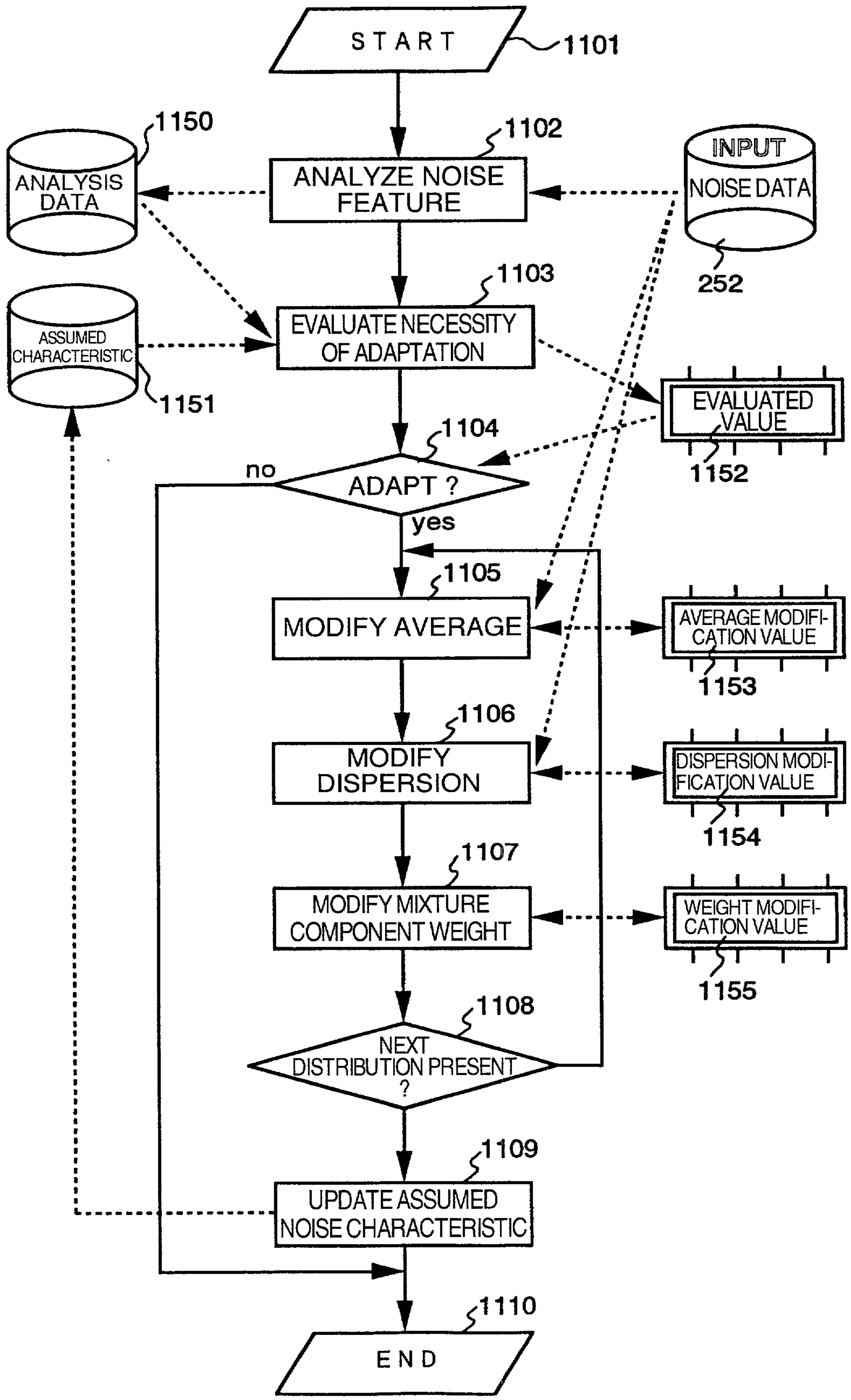


FIG. 20

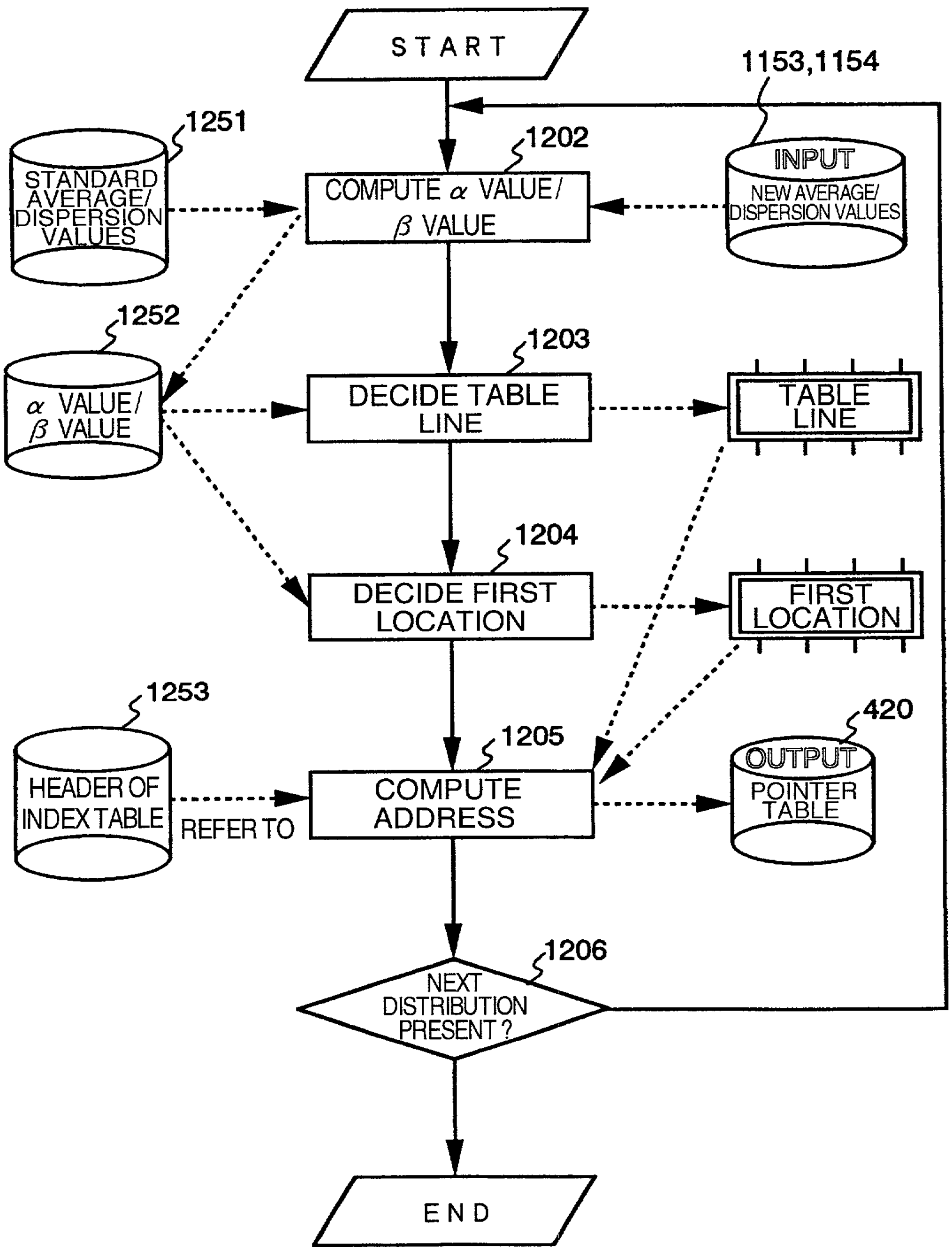

START
1153,1154
1251
1202
STANDARD AVERAGE/ DISPERSION VALUES
COMPUTE α VALUE/ β VALUE
INPUT
NEW AVERAGE/ DISPERSION VALUES
1252
1203
α VALUE/ β VALUE
DECIDE TABLE LINE
TABLE LINE
1204
DECIDE FIRST LOCATION
FIRST LOCATION
1253
420
1205
HEADER OF INDEX TABLE
REFER TO
COMPUTE ADDRESS
OUTPUT
POINTER TABLE
1206
NEXT DISTRIBUTION PRESENT ?
END

FIG. 21

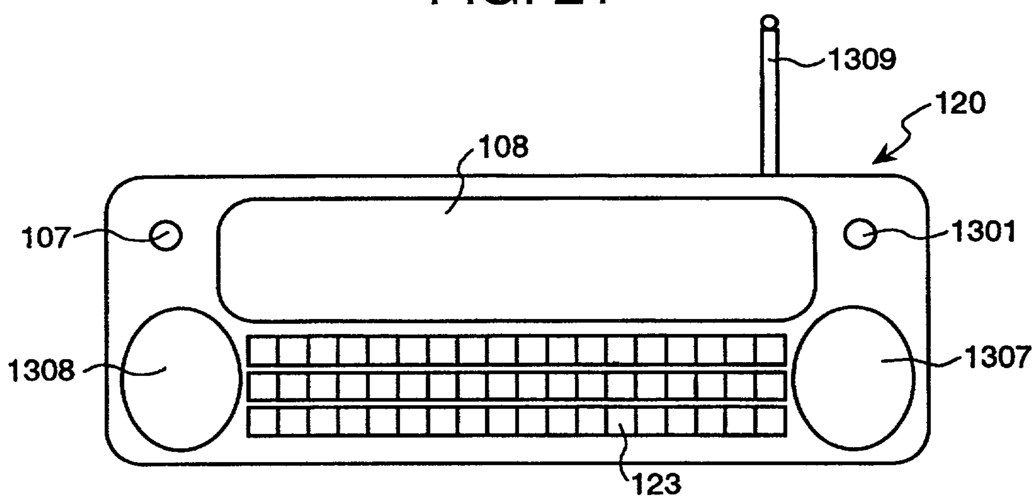

1309
120
108
107
1301
1308
1307
123

FIG. 22

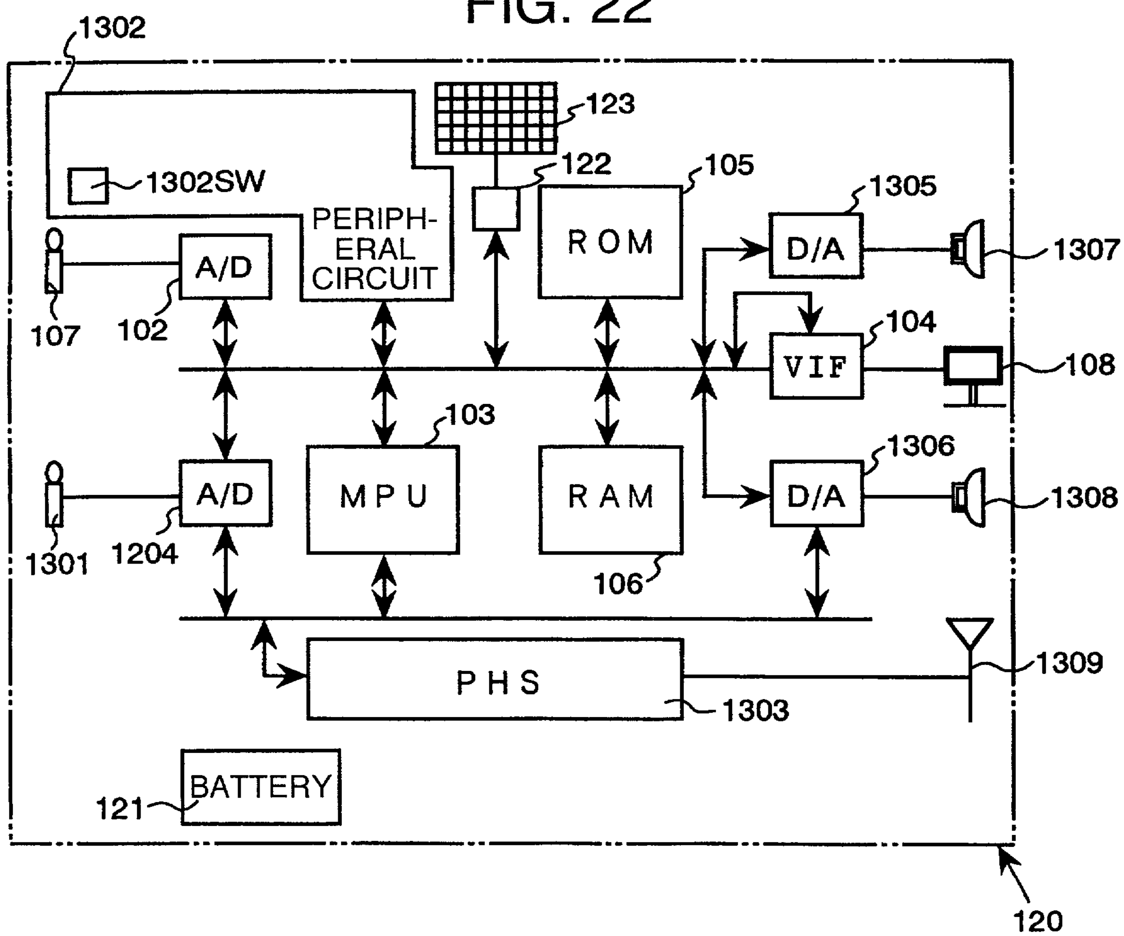

1302
123
1302SW
122
105
1305
PERIPH-
ERAL
CIRCUIT
ROM
D/A
1307
A/D
107
102
104
VIF
108
103
1306
A/D
MPU
RAM
D/A
1308
1301
1204
106
PHS
1309
1303
BATTERY
121
120

FIG. 23

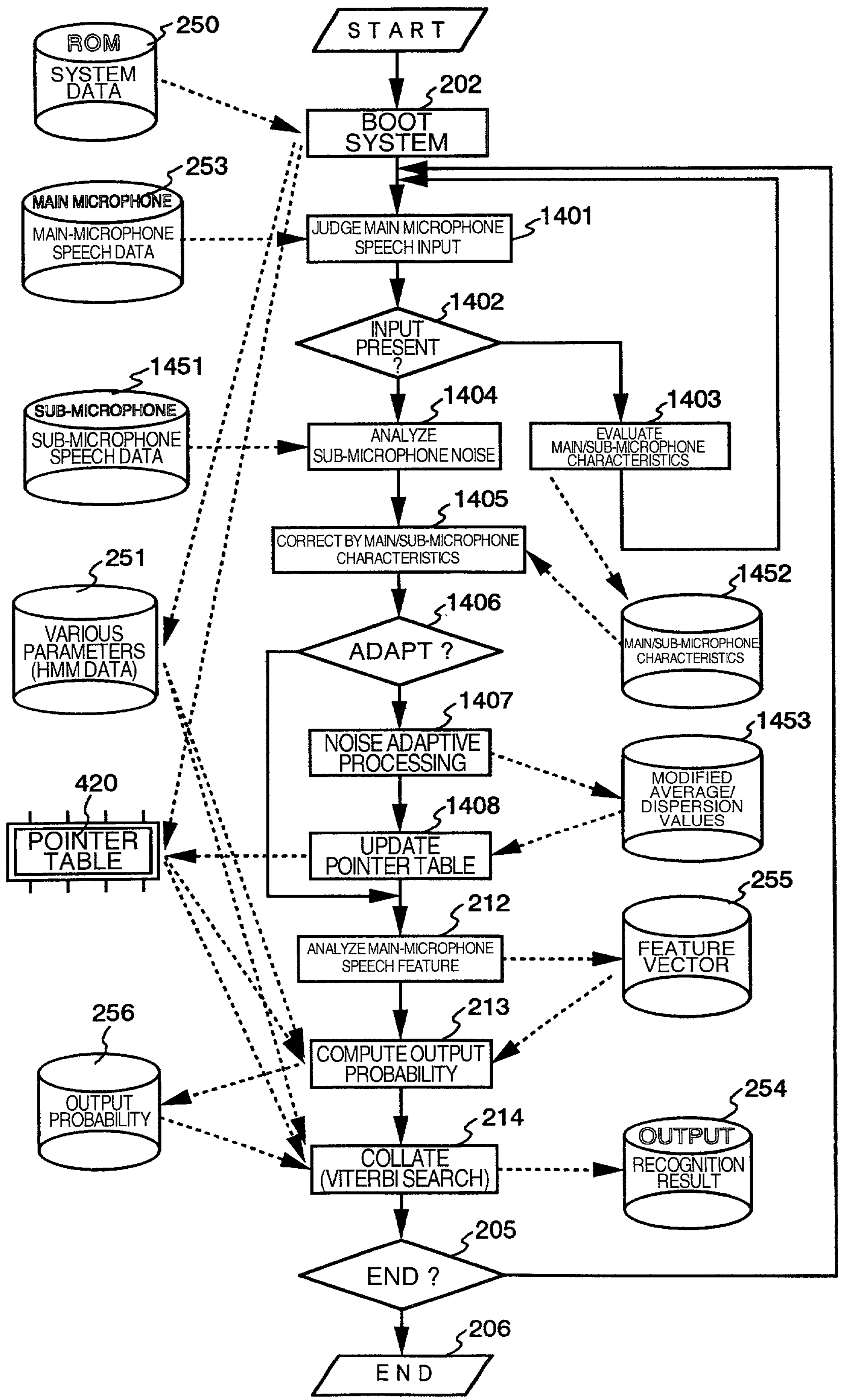
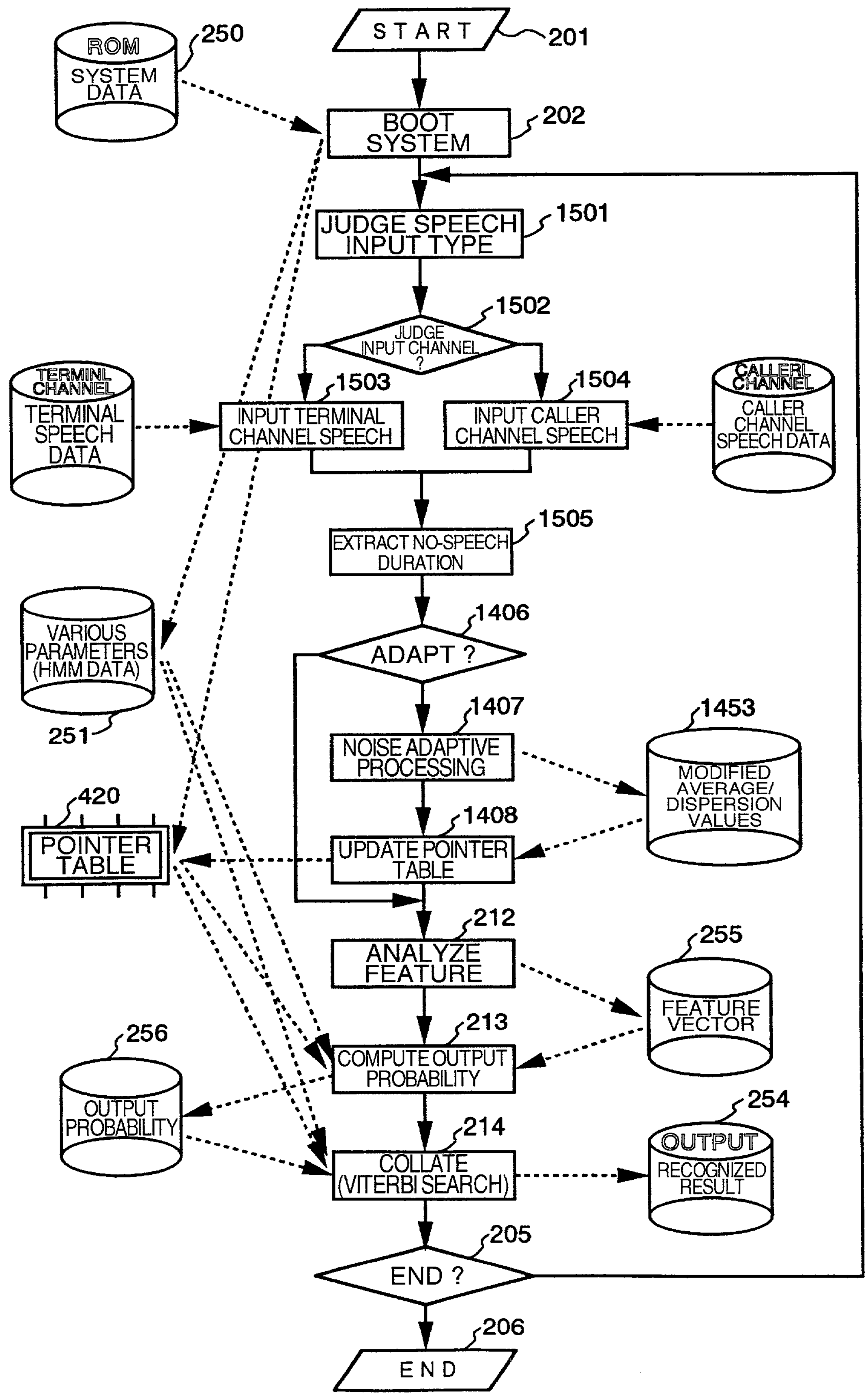

ROM
SYSTEM DATA
250
START
202
BOOT SYSTEM
253
MAIN MICROPHONE
MAIN-MICROPHONE SPEECH DATA
1401
JUDGE MAIN MICROPHONE SPEECH INPUT
1402
INPUT PRESENT ?
1451
SUB-MICROPHONE
SUB-MICROPHONE SPEECH DATA
1404
ANALYZE SUB-MICROPHONE NOISE
1403
EVALUATE MAIN/SUB-MICROPHONE CHARACTERISTICS
1405
CORRECT BY MAIN/SUB-MICROPHONE CHARACTERISTICS
251
VARIOUS PARAMETERS (HMM DATA)
1452
MAIN/SUB-MICROPHONE CHARACTERISTICS
1406
ADAPT ?
1407
NOISE ADAPTIVE PROCESSING
1453
MODIFIED AVERAGE/ DISPERSION VALUES
420
POINTER TABLE
1408
UPDATE POINTER TABLE
212
ANALYZE MAIN-MICROPHONE SPEECH FEATURE
255
FEATURE VECTOR
213
COMPUTE OUTPUT PROBABILITY
256
OUTPUT PROBABILITY
214
COLLATE (VITERBI SEARCH)
254
OUTPUT
RECOGNITION RESULT
205
END ?
206
END

FIG. 24



START
201
ROM
SYSTEM DATA
250
BOOT SYSTEM
202
JUDGE SPEECH INPUT TYPE
1501
JUDGE INPUT CHANNEL ?
1502
TERMINL CHANNEL
TERMINAL SPEECH DATA
INPUT TERMINAL CHANNEL SPEECH
1503
INPUT CALLER CHANNEL SPEECH
1504
CALLERL CHANNEL
CALLER CHANNEL SPEECH DATA
EXTRACT NO-SPEECH DURATION
1505
VARIOUS PARAMETERS (HMM DATA)
251
ADAPT ?
1406
NOISE ADAPTIVE PROCESSING
1407
1453
MODIFIED AVERAGE/ DISPERSION VALUES
420
POINTER TABLE
UPDATE POINTER TABLE
1408
ANALYZE FEATURE
212
255
FEATURE VECTOR
256
OUTPUT PROBABILITY
COMPUTE OUTPUT PROBABILITY
213
COLLATE (VITERBI SEARCH)
214
254
OUTPUT
RECOGNIZED RESULT
END ?
205
END
206

FIG. 25

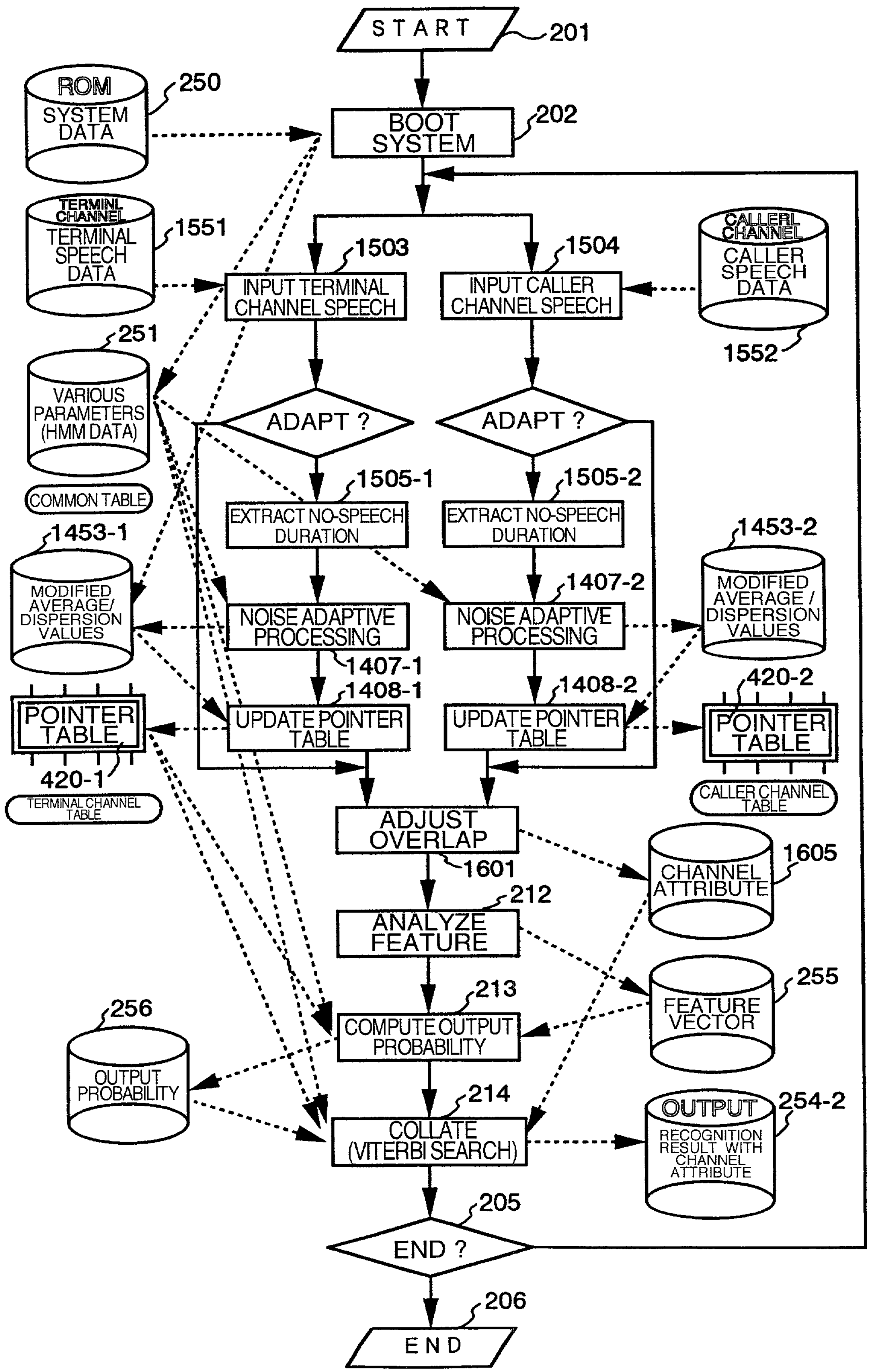
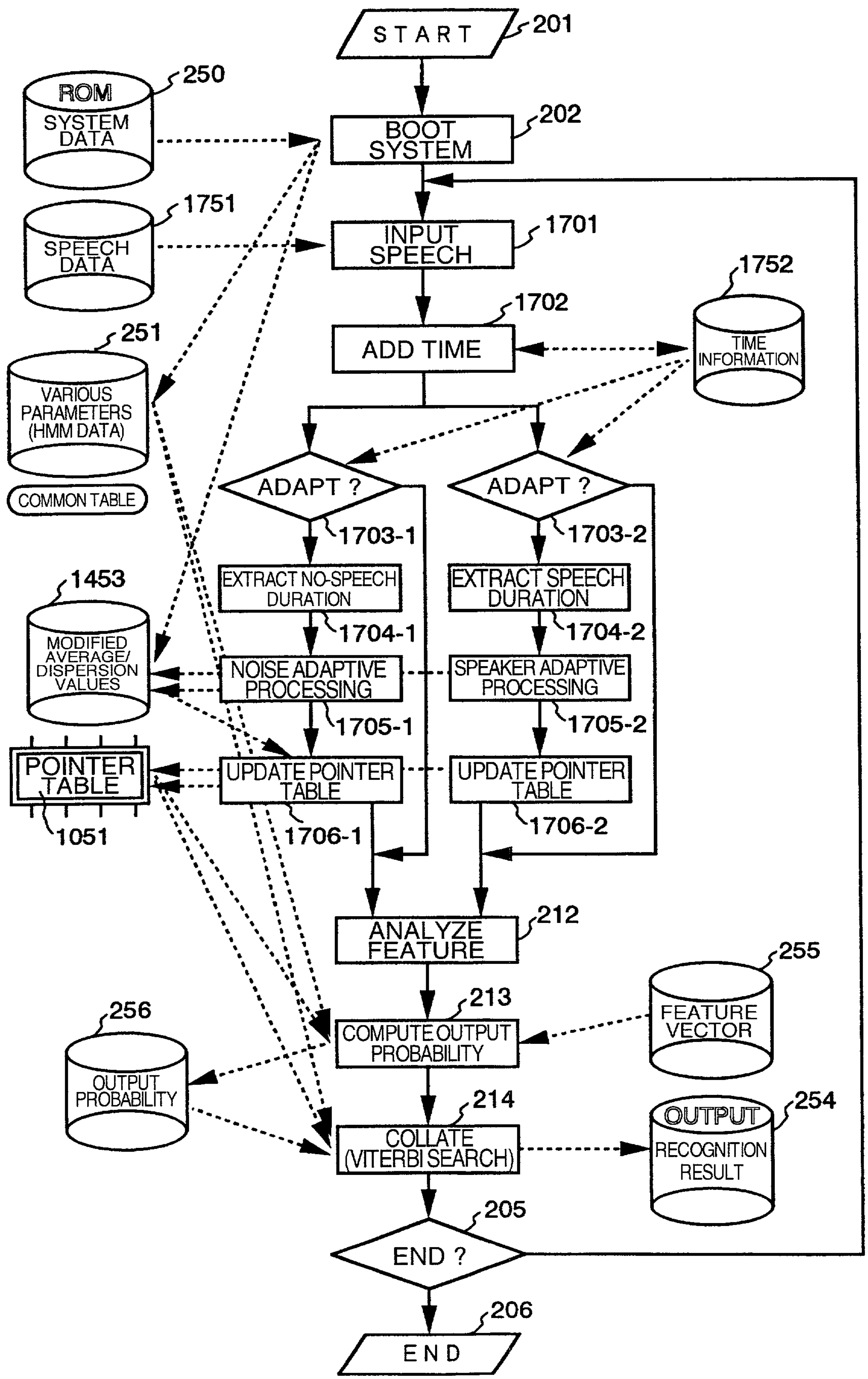
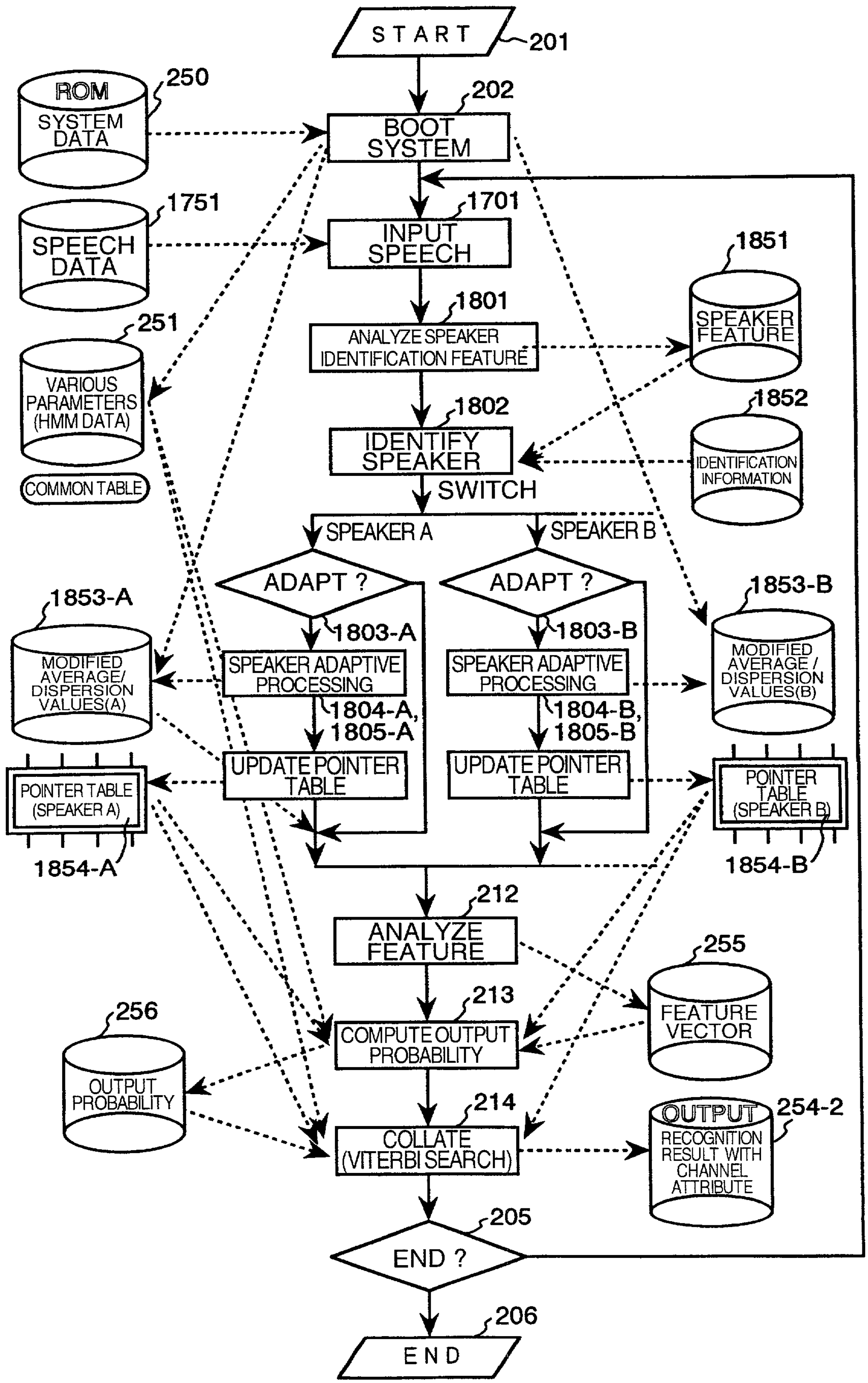

START
201
ROM
SYSTEM DATA
250
BOOT SYSTEM
202
TERMINL CHANNEL
TERMINAL SPEECH DATA
1551
INPUT TERMINAL CHANNEL SPEECH
1503
INPUT CALLER CHANNEL SPEECH
1504
CALLERL CHANNEL
CALLER SPEECH DATA
1552
251
VARIOUS PARAMETERS (HMM DATA)
COMMON TABLE
ADAPT ?
ADAPT ?
1505-1
1505-2
EXTRACT NO-SPEECH DURATION
EXTRACT NO-SPEECH DURATION
1453-1
MODIFIED AVERAGE/ DISPERSION VALUES
1453-2
MODIFIED AVERAGE / DISPERSION VALUES
NOISE ADAPTIVE PROCESSING
NOISE ADAPTIVE PROCESSING
1407-1
1407-2
1408-1
1408-2
UPDATE POINTER TABLE
UPDATE POINTER TABLE
POINTER TABLE
420-1
TERMINAL CHANNEL TABLE
POINTER TABLE
420-2
CALLER CHANNEL TABLE
ADJUST OVERLAP
1601
CHANNEL ATTRIBUTE
1605
ANALYZE FEATURE
212
FEATURE VECTOR
255
COMPUTE OUTPUT PROBABILITY
213
256
OUTPUT PROBABILITY
COLLATE (VITERBI SEARCH)
214
OUTPUT
RECOGNITION RESULT WITH CHANNEL ATTRIBUTE
254-2
END ?
205
END
206

FIG. 26



START
201
250
ROM
SYSTEM DATA
BOOT SYSTEM
202
1751
SPEECH DATA
INPUT SPEECH
1701
1752
TIME INFORMATION
1702
ADD TIME
251
VARIOUS PARAMETERS (HMM DATA)
COMMON TABLE
ADAPT ?
ADAPT ?
1703-1
1703-2
EXTRACT NO-SPEECH DURATION
EXTRACT SPEECH DURATION
1704-1
1704-2
1453
MODIFIED AVERAGE/ DISPERSION VALUES
NOISE ADAPTIVE PROCESSING
SPEAKER ADAPTIVE PROCESSING
1705-1
1705-2
POINTER TABLE
UPDATE POINTER TABLE
UPDATE POINTER TABLE
1051
1706-1
1706-2
ANALYZE FEATURE
212
255
FEATURE VECTOR
256
OUTPUT PROBABILITY
COMPUTE OUTPUT PROBABILITY
213
214
COLLATE (VITERBI SEARCH)
OUTPUT
RECOGNITION RESULT
254
205
END ?
206
END

FIG. 27



START
201
ROM
SYSTEM DATA
250
BOOT SYSTEM
202
SPEECH DATA
1751
INPUT SPEECH
1701
ANALYZE SPEAKER IDENTIFICATION FEATURE
1801
SPEAKER FEATURE
1851
VARIOUS PARAMETERS (HMM DATA)
251
COMMON TABLE
IDENTIFY SPEAKER
1802
IDENTIFICATION INFORMATION
1852
SWITCH
SPEAKER A
SPEAKER B
ADAPT ?
1803-A
ADAPT ?
1803-B
MODIFIED AVERAGE/ DISPERSION VALUES(A)
1853-A
SPEAKER ADAPTIVE PROCESSING
1804-A, 1805-A
SPEAKER ADAPTIVE PROCESSING
1804-B, 1805-B
MODIFIED AVERAGE / DISPERSION VALUES(B)
1853-B
POINTER TABLE (SPEAKER A)
1854-A
UPDATE POINTER TABLE
UPDATE POINTER TABLE
POINTER TABLE (SPEAKER B)
1854-B
ANALYZE FEATURE
212
FEATURE VECTOR
255
COMPUTE OUTPUT PROBABILITY
213
OUTPUT PROBABILITY
256
COLLATE (VITERBI SEARCH)
214
OUTPUT
RECOGNITION RESULT WITH CHANNEL ATTRIBUTE
254-2
END ?
205
END
206

09/890703

FIG. 28

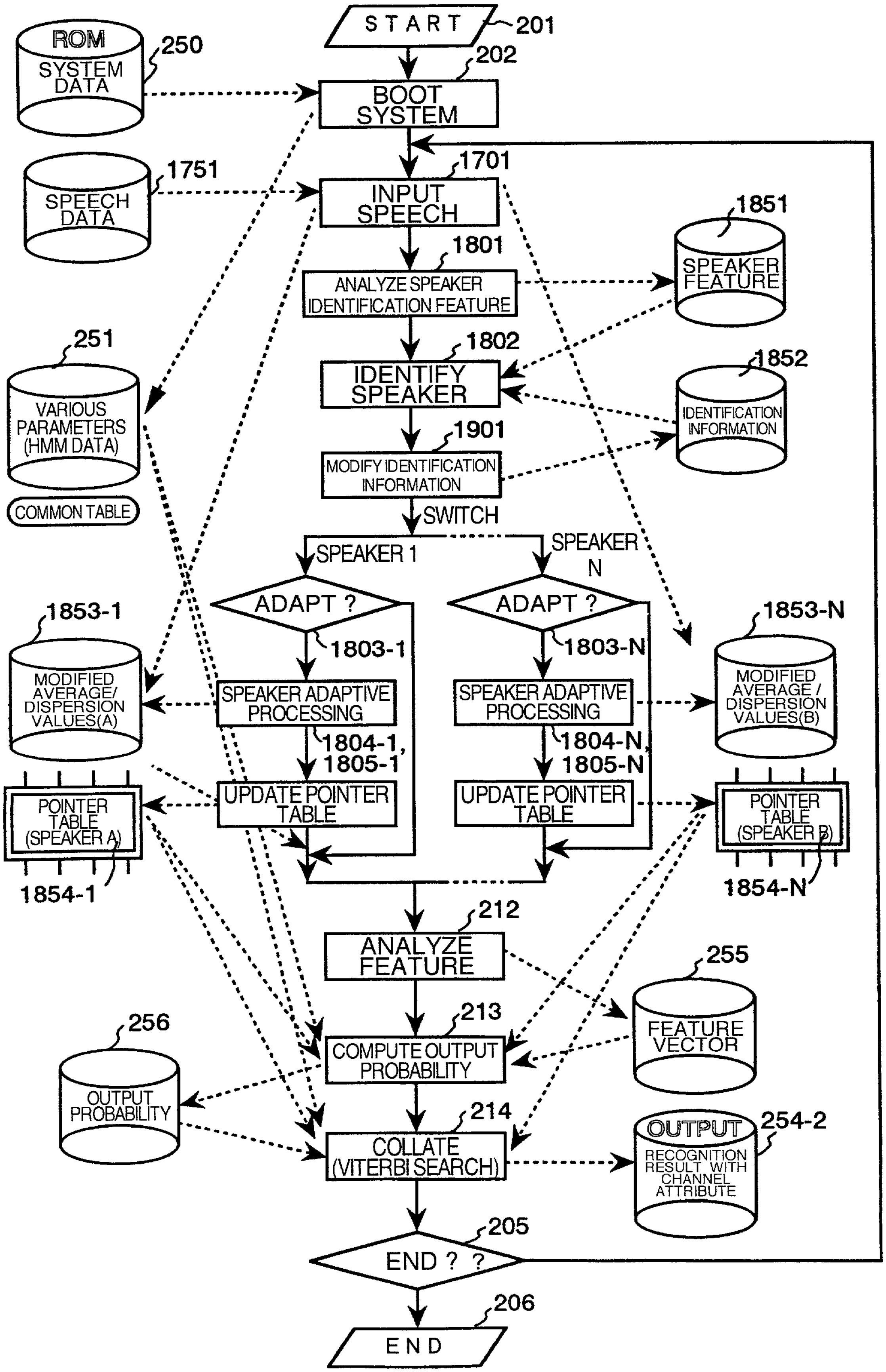


FIG. 29

501 502 503

REGISTERED SPEAKER	FREQUENCY INFORMATION	DATA POINTER
No.5	15	-------
No.2	13	-------
-------	-------	-------
No.4	2	-------

500

FIG. 30

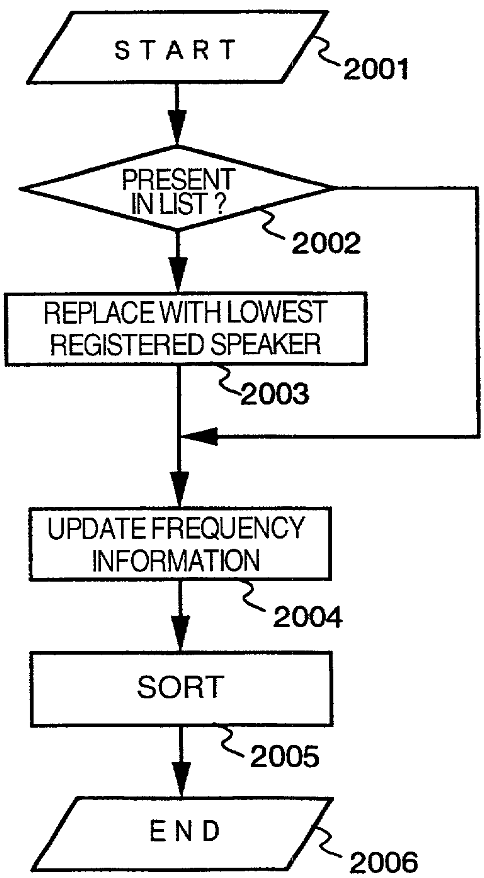


FIG. 31

BUBBLE SORT FROM LOWEREST LINE

REGISTERED SPEAKER	FREQUENCY INFORMATION	DATA POINTER
No.5	15	-------
No.2	13	-------
----------	----------	----------
No.4	2	-------

500

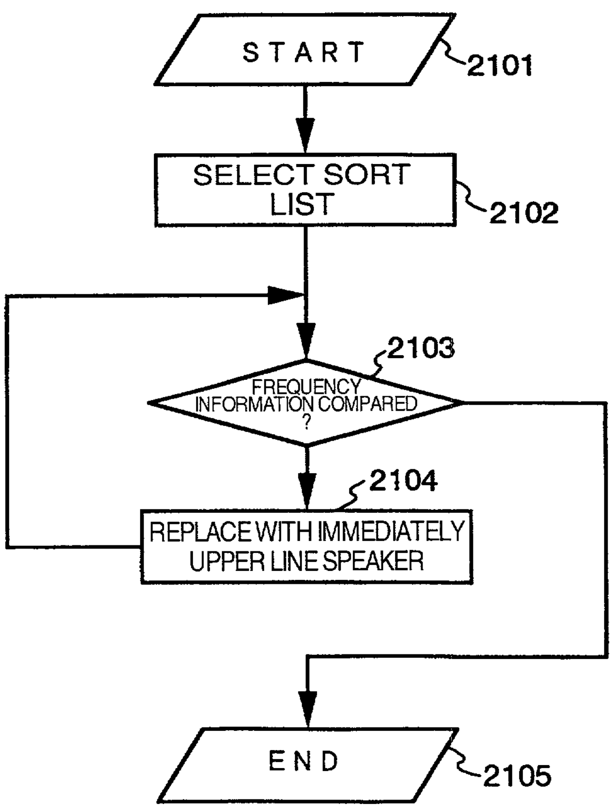
FIG. 32

BUBBLE SORT FROM LIST-PRESENCE LINE

REGISTERED SPEAKER	FREQUENCY INFORMATION	DATA POINTER
No.5	15	-------
No.2	13	-------
----------	----------	----------
No.4	2	-------

500

FIG. 33



09/890703

FIG. 34

<CONCEPT OF TWO-MICROPHONE TYPE NOISE ADAPTIVE PROCESSING>

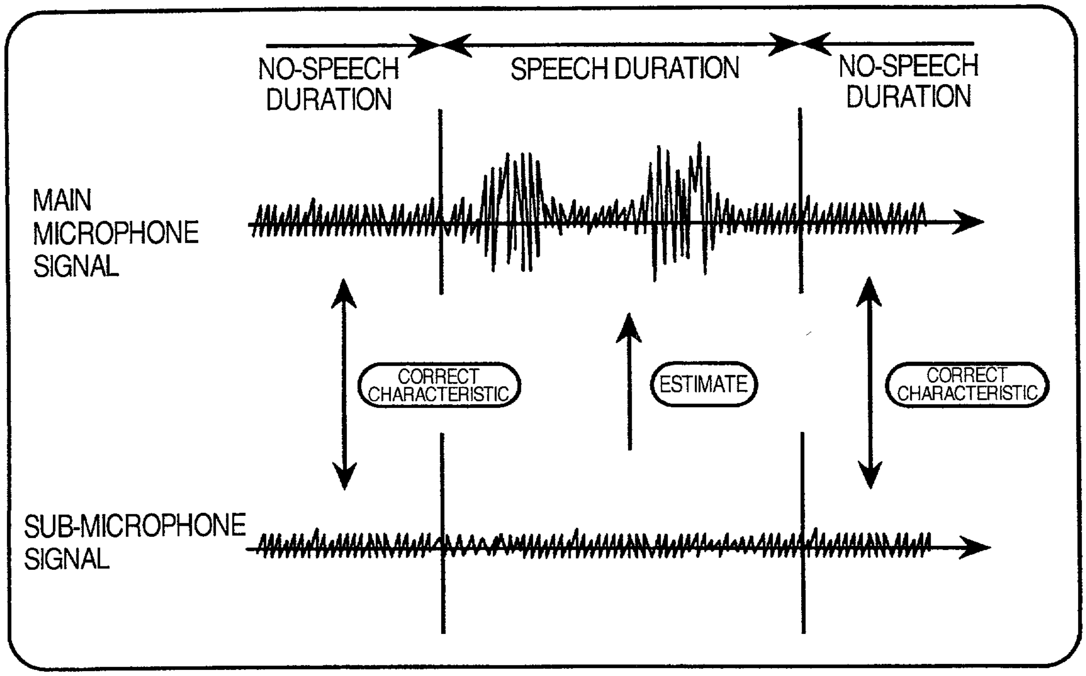


FIG. 35

< SPEECH RECOGNITION IN TRANSCEIVER TYPE SPEECH >

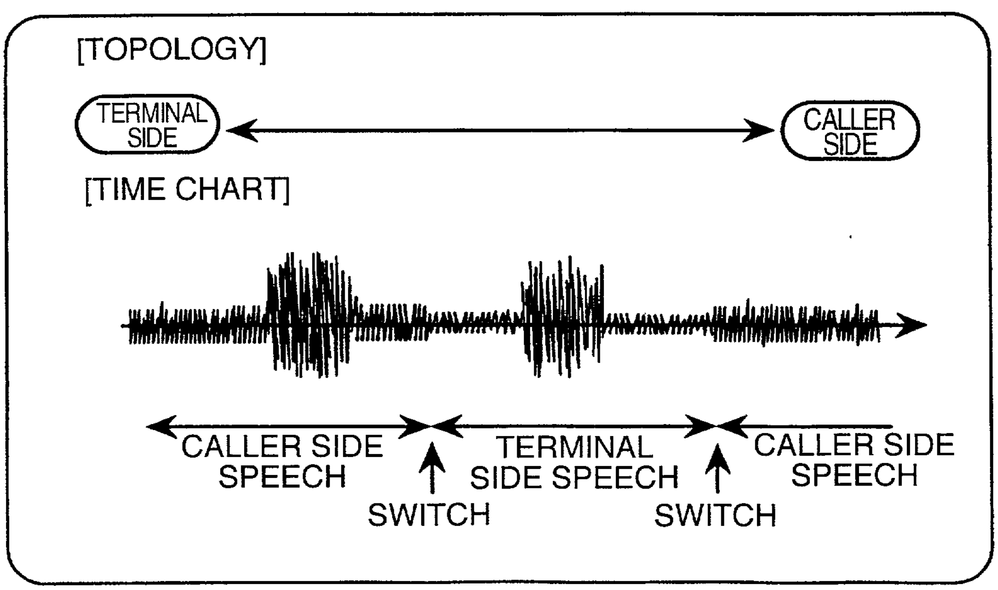


FIG. 36

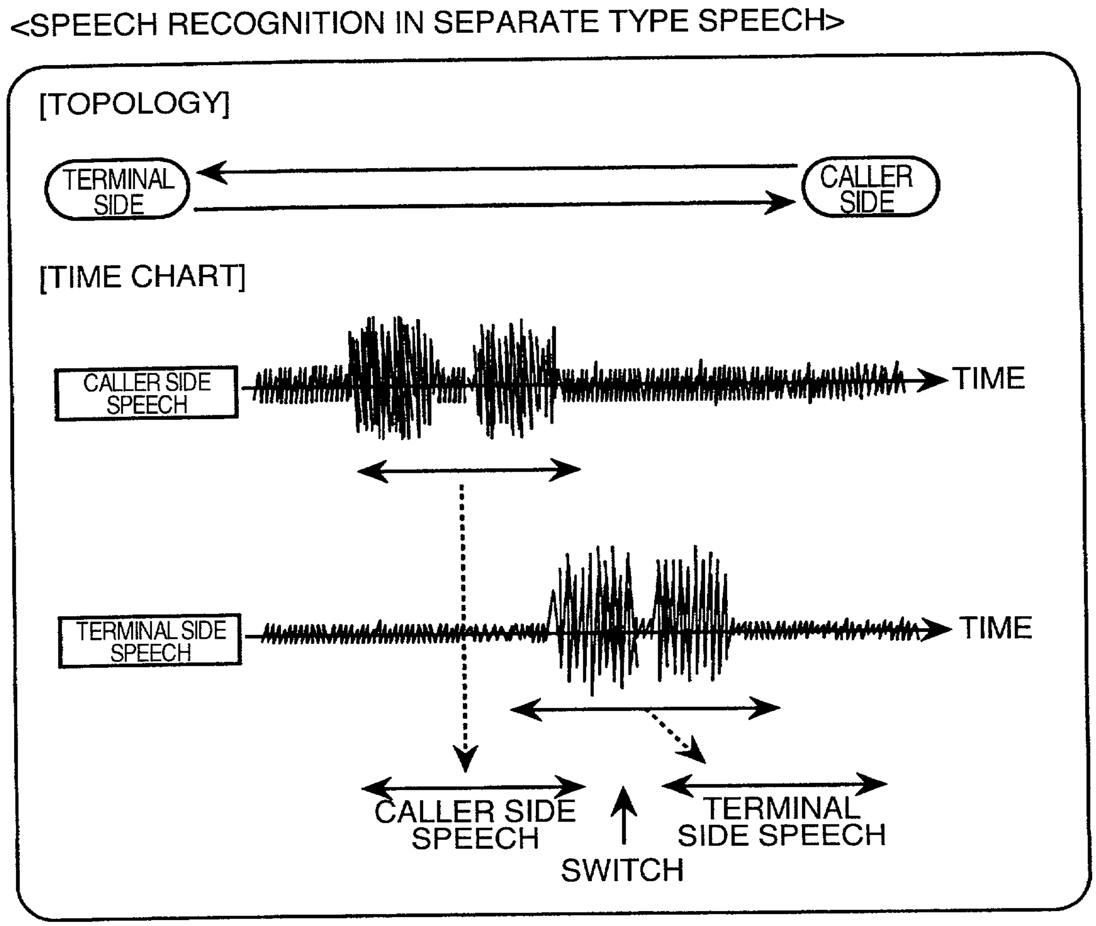

<SPEECH RECOGNITION IN SEPARATE TYPE SPEECH>
[TOPOLOGY]
TERMINAL SIDE
CALLER SIDE
[TIME CHART]
CALLER SIDE SPEECH
TIME
TERMINAL SIDE SPEECH
TIME
CALLER SIDE SPEECH
TERMINAL SIDE SPEECH
SWITCH

FIG. 37

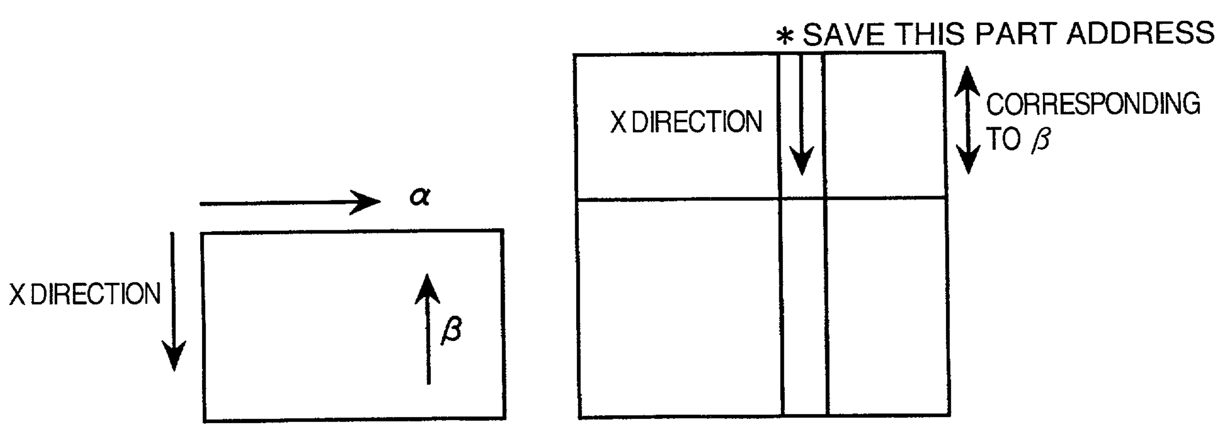
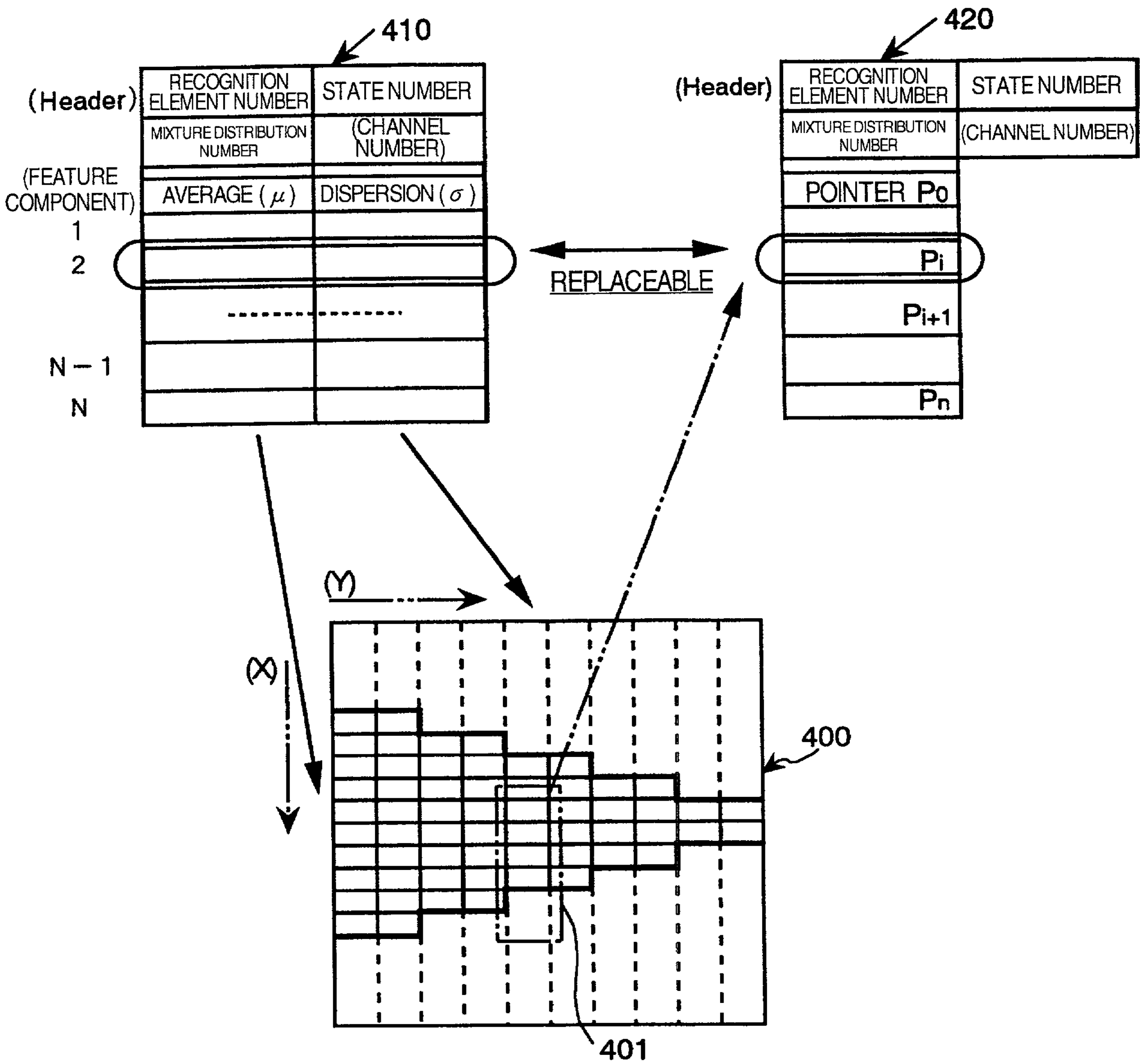
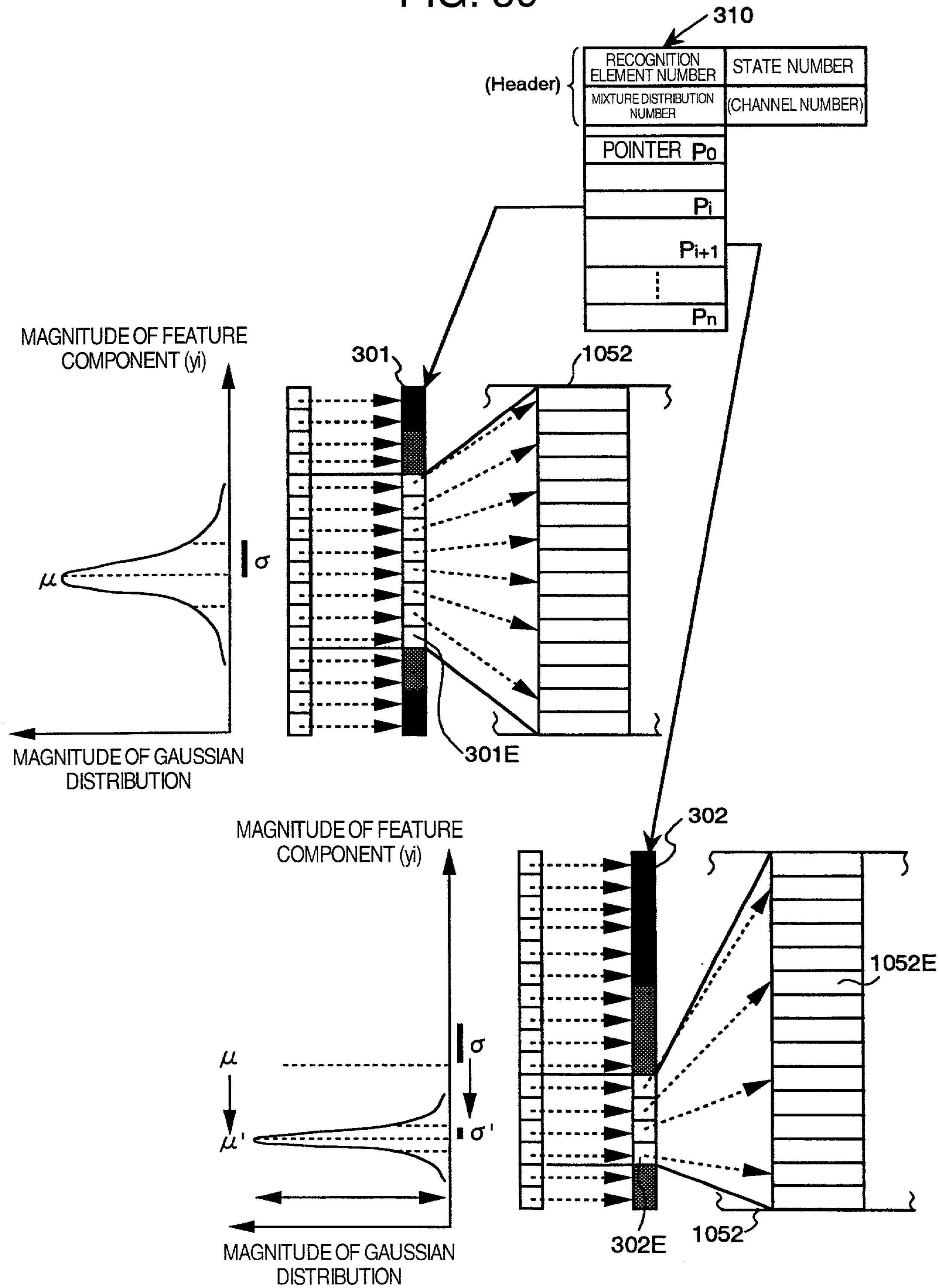

* SAVE THIS PART ADDRESS
X DIRECTION
CORRESPONDING TO β
α
X DIRECTION
β

FIG. 38



P IS UNIQUELY DETERMINED SET (μ , σ)
SET (μ , σ) IS UNIQUELY DETERMINED FOR P

FIG. 39



310
(Header)
RECOGNITION ELEMENT NUMBER
STATE NUMBER
MIXTURE DISTRIBUTION NUMBER
(CHANNEL NUMBER)
POINTER P_0
P_i
P_{i+1}
P_n
MAGNITUDE OF FEATURE COMPONENT (yi)
301
1052
σ
μ
MAGNITUDE OF GAUSSIAN DISTRIBUTION
301E
302
MAGNITUDE OF FEATURE COMPONENT (yi)
1052E
μ
σ
μ'
σ'
MAGNITUDE OF GAUSSIAN DISTRIBUTION
302E
1052

FIG. 40

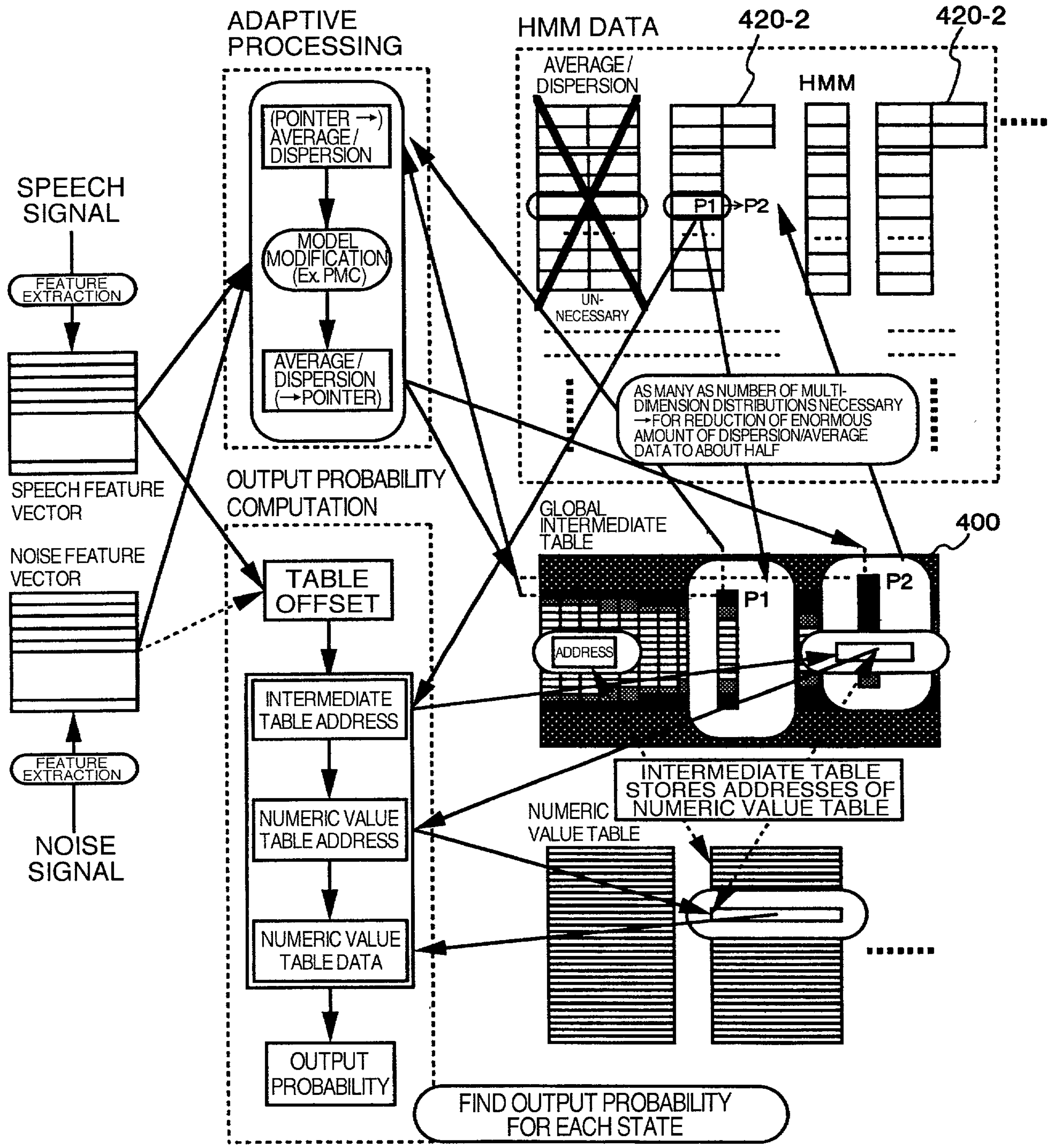

ADAPTIVE PROCESSING
(POINTER →) AVERAGE / DISPERSION
MODEL MODIFICATION (Ex. PMC)
AVERAGE / DISPERSION (→POINTER)
HMM DATA
420-2
420-2
AVERAGE / DISPERSION
HMM
P1
P2
UN-NECESSARY
AS MANY AS NUMBER OF MULTI-DIMENSION DISTRIBUTIONS NECESSARY →FOR REDUCTION OF ENORMOUS AMOUNT OF DISPERSION/AVERAGE DATA TO ABOUT HALF
SPEECH SIGNAL
FEATURE EXTRACTION
SPEECH FEATURE VECTOR
NOISE FEATURE VECTOR
FEATURE EXTRACTION
NOISE SIGNAL
OUTPUT PROBABILITY COMPUTATION
TABLE OFFSET
INTERMEDIATE TABLE ADDRESS
NUMERIC VALUE TABLE ADDRESS
NUMERIC VALUE TABLE DATA
OUTPUT PROBABILITY
GLOBAL INTERMEDIATE TABLE
400
P1
P2
ADDRESS
INTERMEDIATE TABLE STORES ADDRESSES OF NUMERIC VALUE TABLE
NUMERIC VALUE TABLE
FIND OUTPUT PROBABILITY FOR EACH STATE

FIG. 41

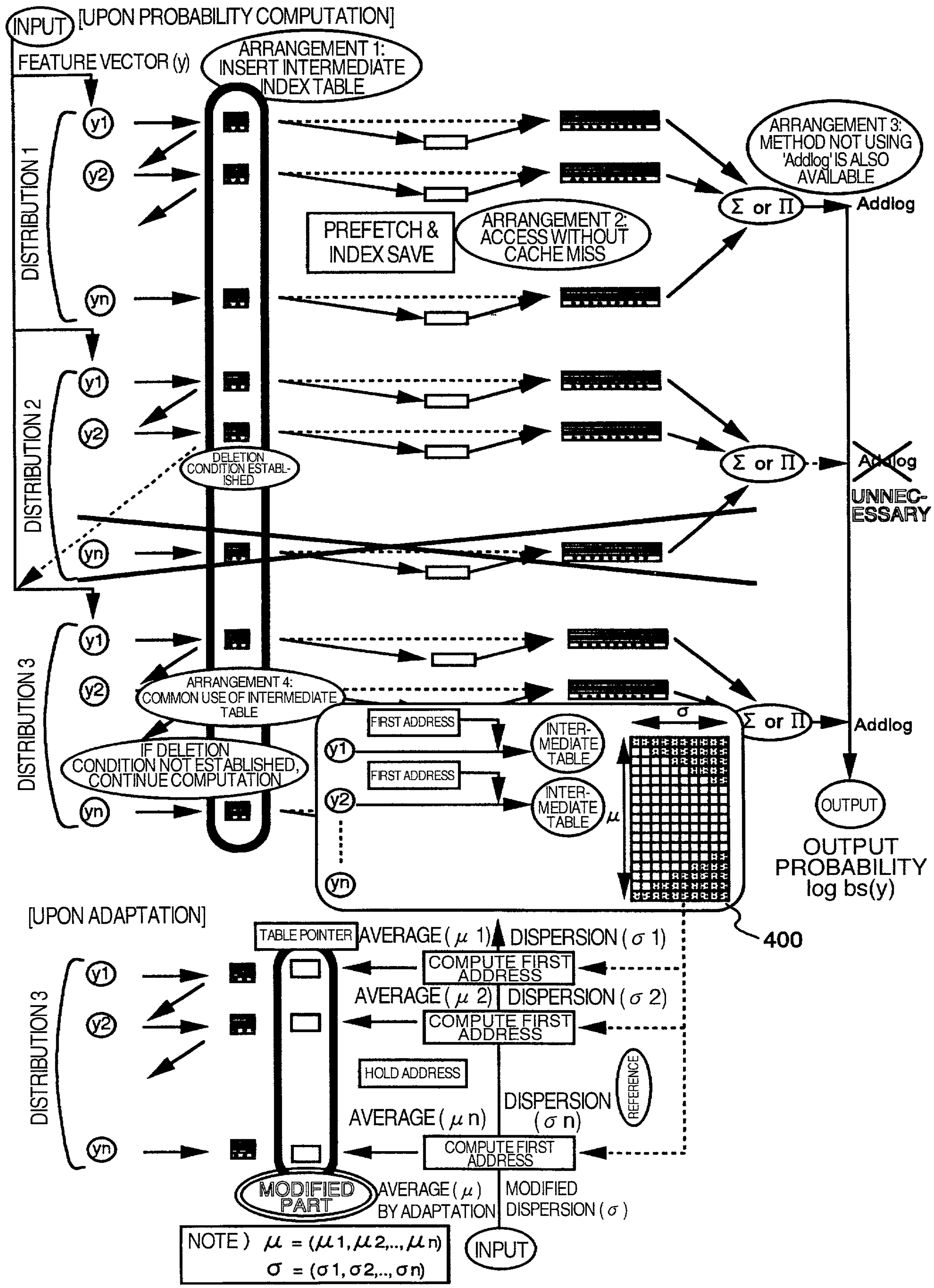

INPUT
[UPON PROBABILITY COMPUTATION]
FEATURE VECTOR (y)
ARRANGEMENT 1: INSERT INTERMEDIATE INDEX TABLE
DISTRIBUTION 1
y1
y2
yn
PREFETCH & INDEX SAVE
ARRANGEMENT 2: ACCESS WITHOUT CACHE MISS
ARRANGEMENT 3: METHOD NOT USING 'Addlog' IS ALSO AVAILABLE
Σ or Π
Addlog
DISTRIBUTION 2
DELETION CONDITION ESTABL-ISHED
Addlog
UNNEC-ESSARY
DISTRIBUTION 3
ARRANGEMENT 4: COMMON USE OF INTERMEDIATE TABLE
IF DELETION CONDITION NOT ESTABLISHED, CONTINUE COMPUTATION
FIRST ADDRESS
INTER-MEDIATE TABLE
σ
μ
OUTPUT
OUTPUT PROBABILITY log bs(y)
400
[UPON ADAPTATION]
TABLE POINTER
AVERAGE (μ 1)
DISPERSION (σ 1)
COMPUTE FIRST ADDRESS
AVERAGE (μ 2)
DISPERSION (σ 2)
HOLD ADDRESS
REFERENCE
AVERAGE (μ n)
DISPERSION (σ n)
MODIFIED PART
AVERAGE (μ) BY ADAPTATION
MODIFIED DISPERSION (σ)
NOTE) μ = (μ1, μ2,.., μn)
σ = (σ1, σ2,.., σn)
INPUT

FIG. 42

[FOR FLOATING POINT OPERATION]

$$\log bs(y) = \log \sum_k \omega_k \prod_i \overset{R1}{\boxed{\frac{1}{\sqrt{(2\pi\sigma_{ski})}} \exp\{-(y_i-\mu_{ski})^2/\sigma_{ski}^2\}}}$$

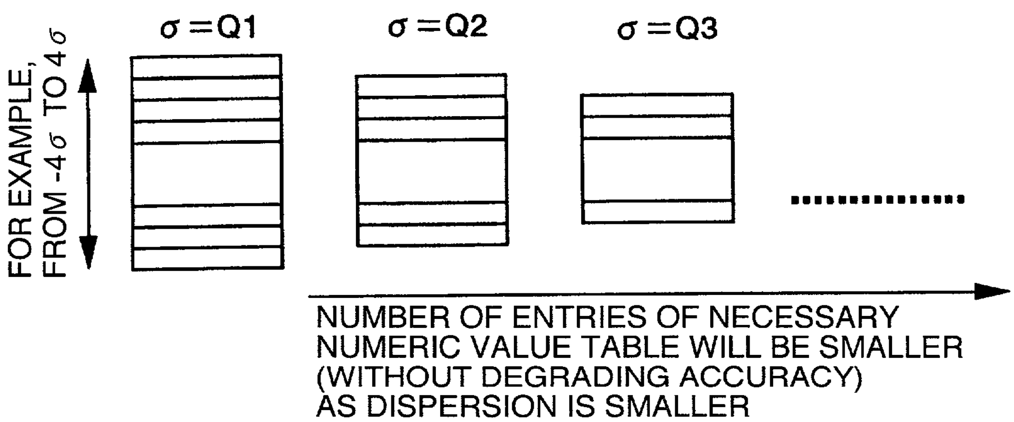


FIG. 43

[CAPABLE OF COPING EVEN WITH INTEGER OPERATION PROCESSING]

NOTE)

$$\log bs(y) = \mathrm{addlog}\{\overset{R3}{\boxed{\log\omega_k}} + \sum_i \overset{R2}{\boxed{\{\{-(y_i-\mu_{ski})^2/\sigma_{ski}^2\} + \log\{\frac{1}{\sqrt{(2\pi\sigma_{ski})}}\}\}}}\}$$

31980 1975

E6055-01 (米)

PTO/SB/106(8-96)
Approved for use through 9/30/98. OMB 0651-0032
Patent and Trademark Office; U.S. DEPARTMETNT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

Declaration and Power of Attorney For Patent Application

特許出願宣言書及び委任状

Japanese Language Declaration

日本語宣言書

下記の氏名の発明者として、私は以下の通り宣言します。

As a below named inventor, I hereby declare that:

私の住所、私書箱、国籍は下記の私の氏名の後に記載された通りです。

My residence, post office address and citizenship are as stated next to my name.

下記の名称の発明に関して請求範囲に記載され、特許出願している発明内容について、私が最初かつ唯一の発明者（下記の氏名が一つの場合）もしくは最初かつ共同発明者であると（下記の名称が複数の場合）信じています。

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

DATA PROCESSING SYSTEM

上記発明の明細書（下記の欄で×印がついていない場合は、本書に添付）は、

The specification of which is attached hereto unless the following box is checked:

☐ __月__日に提出され、米国出願番号または特許協定条約国際出願番号を____________とし、(該当する場合) ____________に訂正されました。

☒ was filed on February 5, 1999 as United States Application Number or PCT International Application Number PCT/JP99/00493 and was amended on ____________________(if applicable).

私は、特許請求範囲を含む上記訂正後の明細書を検討し、内容を理解していることをここに表明します。

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

私は、連邦規則法典第３７編第１条５６項に定義されるとおり、特許資格の有無について重要な情報を開示する義務があることを認めます。

I acknowledge the duty to disclose information which is material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56.

E6055-01(米)

Japanese Language Declaration
(日本語宣言書)

私は、米国法典第３５編１１９条 (a) - (d) 項又は３６５条(b)項に基き下記の、米国以外の国の少なくとも一カ国を指定している特許協力条約３６５ (a) 項に基ずく国際出願、又は外国での特許出願もしくは発明者証の出願についての外国優先権をここに主張するとともに、優先権を主張している、本出願の前に出願された特許または発明者証の外国出願を以下に、枠内をマークすることで、示している。

I hereby claim foreign priority under Title 35, United States Code, Section 119 (a)-(d) or 365(b) of any foreign application(s) for patent or inventor's certificate, or 365(a) of any PCT international application which designated at least one country other than the United States, listed below and have also identified below, by checking the box, any foreign application for patent or inventor's certificate, or PCT International application having a filing date before that of the application on which priority is claimed.

Prior Foreign Application(s)
外国での先行出願

			Priority Not Claimed 優先権主張なし
(Number) (番号)	(Country) (国名)	(Day/Month/Year Filed) (出願年月日)	☐
(Number) (番号)	(Country) (国名)	(Day/Month/Year Filed) (出願年月日)	☐

私は、第３５編米国法典１１９条 (e) 項に基いて下記の米国特許出願規定に記載された権利をここに主張いたします。

I hereby claim the benefit under Title 35, United States Code, Section 119(e) of any United States provisional application(s) listed below.

(Application No.) (出願番号)	(Filing Date) (出願日)	(Application No.) (出願番号)	(Filing Date) (出願日)

私は、下記の米国法典第３５編１２０条に基いて下記の米国特許出願に記載された権利、又は米国を指定している特許協力条約３６５条 (c) に基ずく権利をここに主張します。また、本出願の各請求範囲の内容が米国法典第３５編１１２条第１項又は特許協力条約で規定された方法で先行する米国特許出願に開示されていない限り、その先行米国出願書提出日以降で本出願書の日本国内または特許協力条約国際提出日までの期間中に入手された、連邦規則法典第３７編１条５６項で定義された特許資格の有無に関する重要な情報について開示義務があることを認識しています。

I hereby claim the benefit under Title 35, United States Code, Section 120 of any United States application(s), or 365(c) of any PCT international application designating the United States, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT International application in the manner provided by the first paragraph of Title 35, United States Code Section 112, I acknowledge the duty to disclose information which is material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56 which became available between the filing date of the prior application and the national or PCT international filing date of application.

(Application No.) (出願番号)	(Filing Date) (出願日)	(Status: Patented, Pending, Abandoned) (現況：特許許可済、係属中、放棄済)
(Application No.) (出願番号)	(Filing Date) (出願日)	(Status: Patented, Pending, Abandoned) (現況：特許許可済、係属中、放棄済)

私は、私自身の知識に基ずいて本宣言書中で私が行なう表明が真実であり、かつ私の入手した情報と私の信じるところに基ずく表明が全て真実であると信じていること、さらに故意になされた虚偽の表明及びそれと同等の行為は米国法典第１８編第１００１条に基ずき、罰金または拘禁、もしくはその両方により処罰されること、そしてそのような故意による虚偽の声明を行なえば、出願した、又は既に許可された特許の有効性が失われることを認識し、よってここに上記のごとく宣誓を致します。

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

E6055-01 等)

Japanese Language Declaration
(日本語宣言書)

委任状： 私は下記の発明者として、本出願に関する一切の手続きを米特許商標局に対して遂行する弁理士または代理人として、下記の者を指名いたします。（弁護士、または代理人の氏名及び登録番号を明記のこと）	POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith (list name and registration number) John R. Mattingly, Reg. No.30,293; Daniel J. Stanger, Reg. No.32,846; Shrinath Malur, Reg.No.34,663; Gene W. Stockman, Reg. No.21,021; Jeffrey M. Ketchum, Reg. No.31,174; and Scott W. Brickner, Reg. No.34,553.
書類送付先	Send Correspondence to: Mattingly, Stanger & Malur, P.C 104 East Hume Avenue Alexandria, Virginia 22301
直接電話連絡先：（氏名及び電話番号）	Direct Telephone Calls to: (*name and telephone number*) Telephone: (703) 684-1120 Fax (703) 684-1157

唯一または第一発明者	Full name of sole or first inventor Motohito NAKAGAWA
発明者の署名 日付	Inventor's signature Date Motohito Nakagawa 12/July/2001
住所	Residence Chiba, Japan JPX
国籍	Citizenship Japan
私書箱	Post Office Address c/o Hitachi, Ltd., Intellectual Property Group New Marunouchi Bldg. 5-1, Marunouchi 1-chome, Chiyoda-ku, Tokyo 100-8220, Japan

（第二以降の共同発明者についても同様に記載し、署名をすること）

(Supply similar information and signature for second and subsequent joint inventors.)

E6055-01 (米)

PTO/SB/106(8-96)
Approved for use through 9/30/98. OMB 0651-0032
Patent and Trademark Office; U.S. DEPARTMETNT OF COMMERCE

280

第二共同発明者	Full name of second joint inventor, if any Hideo MAEJIMA
第二共同発明者の署名　日付	Second inventor's signature　Date Hideo Maejima　2/July/2001
住所	Residence Hitachi, Japan　JPX
国籍	Citizenship Japan
私書箱	Post Office Address c/o Hitachi, Ltd., Intellectual Property Group New Marunouchi Bldg. 5-1, Marunouchi 1-chome, Chiyoda-ku, Tokyo 100-8220, Japan
第三共同発明者	Full name of third joint inventor, if any
第三共同発明者の署名　日付	Third inventor's signature　Date
住所	Residence
国籍	Citizenship
私書箱	Post Office Address
第四共同発明者	Full name of fourth joint inventor, if any
第四共同発明者の署名　日付	Fourth inventor's signature　Date
住所	Residence
国籍	Citizenship
私書箱	Post Office Address
第五共同発明者	Full name of fifth joint inventor, if any
第五共同発明者の署名　日付	Fifth inventor's signature　Date
住所	Residence
国籍	Citizenship
私書箱	Post Office Address